COMPUTER SELECTION AND EVALUATION

A. Nevzat Bayraktar

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

COMPUTER SELECTION AND EVALUATION

by

A. Nevzat Bayraktar

June 1978

Thesis Advisor: Ronald J. Roland

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) Computer Selection and Evaluation	5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; June 1978
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) A. Nevzat Bayraktar	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940	12. REPORT DATE June 1978
	13. NUMBER OF PAGES 211
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, California 93940	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer Selection

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Published approaches to computer evaluation and selection were reviewed. Preliminary steps and specification development were discussed. Alternative techniques for computer evaluation and for workload description were contrasted. Proposed solicitation methods, computer procurement plans and computer performance measurement and evaluation techniques were surveyed.

Computer Selection and Evaluation

by

A. Nevzat Bayraktar
Lieutenant, Turkish Navy
B.S., Turkish Naval Academy, 1971

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
June 1978

ABSTRACT

Published approaches to computer evaluation and selection were reviewed. Preliminary steps and specification development were discussed. Alternative techniques for computer evaluation and for workload description were contrasted. Proposed solicitation methods, computer procurement plans and computer performance measurement and evaluation techniques were surveyed.

TABLE OF CONTENTS

I. INTRODUCTION ---------- 7

II. PRELIMINARY STEPS TO SELECTION ---------- 12

III. SPECIFICATION DEVELOPMENT ---------- 16

A. GENERAL SPECIFICATIONS ---------- 17

B. DETAILED SPECIFICATIONS ---------- 19

C. OTHER APPROACHES ---------- 20

IV. LIMITATIONS IMPOSED ON SELECTION ---------- 22

A. MANDATORY REQUIREMENTS ---------- 22

B. DESIRABLE VERSUS MANDATORY REQUIREMENTS ---------- 22

C. LIMITING CONDITIONS ---------- 25

V. PROPOSAL EVALUATION TECHNIQUES ---------- 29

A. TECHNIQUES FOR THE EVALUATION OF COMPUTER SYSTEMS ---------- 33

1. Simple Techniques ---------- 33

a. Sole Source ---------- 33

b. Subjective Judgment (Overall Impression) ---------- 33

c. Cost Only ---------- 34

d. A Case History ---------- 35

2. Sophisticated Techniques ---------- 37

a. Weighted-Scoring Technique ---------- 37

b. Cost/Effectiveness Ratio ---------- 39

c. Cost-Value Technique ---------- 39

d. Requirements-Costing Technique ---------- 65

e. Dynamic Approach ---------- 65

f. Present Value Analysis ---------- 75

VI. SYSTEM WORKLOAD DESCRIPTION---------------------- 78
A. INSTRUCTION MIX-------------------------------- 83
B. BENCHMARK PROGRAMS----------------------------- 87
1. Derivation of Representative Programs--- 91
a. Application-------------------------- 91
b. Programs and Tasks------------------- 91
c. Task Summary------------------------- 93
d. Selection of Representative Tasks--- 96
e. Extension Factors-------------------- 97
f. Mix of Tasks------------------------- 99
2. Expected Workload Levels---------------- 99
VII. METHODS OF PROCURING COMPUTER SYSTEMS----------- 105
A. COMMON PROCUREMENT PLANS-------------------- 105
1. Leasing------------------------------------ 105
a. Lessee Motivation------------------- 114
b. Lessor Motivation------------------- 118
c. Lease Contract---------------------- 120
2. Purchasing-------------------------------- 123
a. Purchase Contract------------------- 125
b. Rental Contract--------------------- 126
3. Lease With Purchase Option-------------- 128
4. Lease To Ownership---------------------- 132
B. LEASING VERSUS PURCHASING------------------- 134
1. Methods for Lease-or-Purchase Decision------------------------------------- 134

VIII. SOLICITING PROPOSALS------------------------------------ 152

A. REQUEST FOR PROPOSAL (RFP)------------------ 154

B. VENDOR CONTACT------------------------------ 159

1. Bidders' Conference and Questions------ 159

2. Vendors' Demonstrations and Presentations------------------------------ 160

3. Contracting and Debriefing------------- 161

IX. COMPUTER PERFORMANCE MEASUREMENT AND EVALUATION-- 164

A. THE SYSTEM PERFORMANCE EVALUATION METHODOLOGY---------------------------------- 171

B. THE CONTROL OF SYSTEM PERFORMANCE---------- 174

C. CLASSIFICATION OF DATA PROCESSING SYSTEMS-- 180

D. UNIT OF DATA PROCESSING WORK--------------- 181

1. Batch Systems---------------------------- 181

2. Real Time Systems----------------------- 184

3. Interactive Time-Sharing Systems------- 184

X. CONCLUSION AND RECOMMENDATIONS----------------- 186

APPENDIX A - PERFORMANCE CRITERIA--------------------- 191

APPENDIX B - HARDWARE CHECKLIST---------------------- 194

APPENDIX C - SOFTWARE CHECKLIST---------------------- 202

BIBLIOGRAPHY-- 206

INITIAL DISTRIBUTION LIST----------------------------- 211

I. INTRODUCTION

In the past twenty years, the computer industry has sustained a technical reevaluation unrivaled in modern history. The computer has become the greatest management tool of our time. Yet its proper application contains many pitfalls, as case after case of dramatic failures testify. One of these dangers is improper equipment selection. When managers thoughtlessly procure equipment as a natural process item, they can easily preclude any possibility of success simply by buying the wrong equipment.

The computer selection and evaluation process has become a complex one, requiring detailed attention; it can involve hundreds of technical as well as nontechnical considerations. Today's computer systems are typically very complex and extensive in composition and operation. Academic and administrative users of computer systems have traditionally left most of the considerations in systems selection to technical personnel. As a result, many user needs have gone unsatisfied. Technicians have become frustrated because they often found out too late, if at all, what user needs and priorities really were. In addition, many technicians have had difficulty in communicating to relatively untrained users a complete understanding of the actual capabilities of various computer systems.

The remarkable technical reevaluation in the industry has led to the creation and ultimate availability of a large

number of unique computer systems. When one considers the vast number of peripheral devices available with these basic systems, added to the various special purpose and analog devices, the number of unique computer configurations is almost infinite. To this confusing marketplace the prospective buyer brings his enthusiasm, but not a disciplined approach to the selection process. Technical progress and new application opportunities occur so fast that any organization's equipment strategy should come under frequent review. There are any number of circumstances that dictate the requirement for an open evaluation of computer requirements.

The competitive nature of the computer industry has caused vast technical changes during the last decade; during this period, the industry has moved from punched card orientation to online communications. Relatively fast memories and large capacity direct access devices with relatively fast access times and sophisticated operating systems, also have been made available with present-generation data processing systems. Large funds, placed in research and development, have led to ever increasing numbers of new systems, each one larger in capacity, faster, more capable, with more software than the last ones. The industry is dynamic. The pressure on the present user to move from his presently obsolete system to the newer, more powerful system is logical and demanding of analysis.

Allied to the technical changes in the data processing field are economic changes. Equipment is now being made available at substantial cost reductions. One can procure

a third-generation system at 40 to 60 per cent of the cost of equivalent older equipment. The user who formerly made changes only when present equipment was incapable of satisfying processing needs is now forced to compare the new added equipments on the basis of technical and economic advantages. The continual changes in the field require such an analysis on a periodic basis [Tatham 1969, Yearsley 1973, Thrussell 1976, Joslin 1977].

The purchase of a computer system is a major consideration for any company, but it is often approached and investigated in a superficial way. Computer salesmen are called in, but in many cases the company does not get maximum usage from a computer purchased from a high pressure salesman. Disillusionment spreads among the users, and it is quite often based on preconceived notions of the purchased computer. It is considered a "save all and do all" type deal. You can save all this money because the computer will do all the tedious work. They fail to realize that even though a computer can produce some material 24 hours a day, it does not mean that the computer is alive. Rather, the computer is dead simply because management is dead to possibilities. The manager should be completely aware of the computer's potential, since he finds himself sitting in on or leading many management teams. He should be aware of the intangible problems that may cause the computer to become economically unfeasible, or which may creep in later if the computer is implemented.

The effectiveness and potential of any computer-based system is strongly influenced by the design of that system as well as the choice of equipment. Thus the scope of the equipment available must be evaluated when selecting the equipment in order to understand how a particular choice affects the planned system.

The organization entering the computer world for the first time normally lacks in-house talent since it may not have any people having background in computer applications. This leads to major and often complete reliance on the marketing wiles and brochuremanship of the computer manufacturer. Like the uninformed buyer in most fields, the uninformed Automatic Data Processing (ADP) user normally contracts with the reputable firm, hoping that the firm is so interested in the user's unique applications that he will get unusual service. Thus, a major decision, capable of affecting the future competitive position of the firm, is quite easily turned over to the equipment manufacturer. A much better solution can be to seek outside consulting help. Even a substantial investment in consulting fees can pale in relation to the cost of a poor selection.

The selection of a computer and manufacturer is one of the major decisions to be made in formulating an organization's computer policy. Also for the following reasons the decision is important; the equipment itself is probably the largest individual expense in the computer department, and

the selection of manufacturer and a specific computer sets a constraint on system development that will last at least five years [Clifton 1969, Tatham 1969, Wooldridge 1973, Procop 1976, Withington 1974, Sabol 1972].

II. PRELIMINARY STEPS TO SELECTION

Once it is clear that evaluation of computer needs is necessary under a particular set of circumstances, a series of studies as preliminary steps to the actual process of selection should be initiated. These steps are time-consuming, but they are an essential tool that managers can use to make an accurate evaluation of the equipment requirements. This study is designed to determine the cost effectiveness of various solutions to the data processing problem. These alternative solutions will include the present system, plus various combinations of data processing systems. For each alternative, there will be a distinct systems approach, as well as an appropriate cost.

Without being too ambitious and without being too restrictive, the planning for a data processing system must look at today but it must also assess tomorrow. Some of these tomorrow's requirements can be reasonably forecast today. Many of them result from growth patterns of the enterprise. Such growth patterns and changes are fundamental in nature and demand a coordinated management information system which can: screen all company data and prepare reports according to requirements, provide faster factual information in a dependable and management oriented manner, help management with both present and future information needs and provide data on events before they happen.

It is important to distinguish between what management wants and what it needs. The comparative analysis will allow management to exercise its judgment as to the feasibility of new system development. It is important to remember that this new system cannot, and should not, be undertaken on the basis of cost savings alone. The value of information, though difficult to quantify, must be considered in the study. The incremental cost between two alternatives will often be more than offset by the value that the increased information will give to the decision-maker. It is also important that these factors be weighed at high levels of the organization's management.

Once an approach has been accepted by management, a macrosystem design effort begins. Information flows are defined, inputs and outputs are determined, and files are developed. Although all the basic data required by the system are defined and gross flow charts developed, the system designer is limited, at this point, by his lack of knowledge of the specific hardware configuration. However, it is during the applications study that the model which will ultimately become the operational system is created. Insufficient effort at this point can only lead to poor equipment selection and development of a stunted system. Now, failure to undergo the detailed information study can only mean postponement to the time when maximum effort must be allotted to software, procedures, and training development.

The importance of the application study cannot be overemphasized. The results of the study are normally appended to the system specifications for the hardware manufacturers to use as a basis for proposals. Because of the importance in defining the model system for vendor proposals, it is at the applications study level that professional data processing support must be made available. Decisions on offline versus online systems, disc versus tape, two-channel versus eight-channel, basic processor speed, and memory size, require highly skilled systems analysis personnel thoroughly familiar with both the state of the art and the function to be automated. Unfortunately, such personnel, almost without exception, are in very short supply [Clifton 1969, Joslin 1977, Tatham 1969, Chorafas 1967].

In addition, the complexity of modern systems is so great that it is almost impossible for a systems analysis team to consider all the major alternatives. Six years computer experience with a company may be an excellent base for systems analysts or data processing management but again unfortunately it seldom covers experience in the process of computer selection or in the scale and technology about to be investigated.

The best solution to the above problem is the use of simulation techniques to assist in the system design. With this technique, a description of the user's workload (files and programs), along with the appropriate hardware and software characteristics of the configuration to be simulated, is used as input to a sophisticated computer program. The

program is a generalized mathematical model of computer processing, which enables the workload to be simulated and valuable performance data to be collected. In this way, the system design analyst can try out different ideas and approaches in order to arrive at a good system design.

Use of the simulation approach can drastically reduce the amount of elapsed time required for the applications study effort, and results in a better system design by providing more analysis than is possible with a manual method. If this approach is taken, caution should be exercised to ensure that the simulation model accurately handles the essentials of third-generation processing. In order to be most useful for system design, the simulator should allow easy man/machine interaction by: fast turnaround time, output results oriented toward suggesting system improvements, and flexibility allowing design or configuration changes to be made easily [Joslin 1977, Graham 1973, Clifton 1969 Martin 1973].

III. SPECIFICATION DEVELOPMENT

The topic of specifications plays an important role in system evaluation. Development of specifications, which will be released to all interested manufacturers, is a crucial part of the process. Specifications for a computer system can be prepared in a number of different ways. The specifications should be general enough to assure wide competition, yet specific enough to delineate the user's requirements clearly. The final result of the application study is a documented model system. This system should become part of, and treated by, the specifications as a point of departure from which each manufacturer is free to use his own ingenuity and brain power to develop a superior system, oriented to his own equipment. Proper control of this process is maintained by establishing the constraints within which each manufacturer must work.

The design of the system specifications should be the starting place for developing any computer selection plan. It should define what is sought in the way of a computer system, give the system requirements for the various applications, and give a detailed description of each step of each application. The system specifications reflect the findings of the system study team. The final choice can be detrimental to both the company and the vendors if proper care is not taken in the preparation of the specifications.

The specifications must reflect the actual applications to be handled by the system, and must not contain poorly thought out limitations which are to be imposed on the system. Also they should not be directed too much toward a specific systems approach [Joslin 1977, Chorafas 1967, Wooldridge 1973]. The several methods are described in the following sections.

A. GENERAL SPECIFICATIONS

General specifications are nothing more than the findings of the analysis, a description of the jobs to be done: the inputs, the desired outputs, and any other pertinent parameters. (As an example of outputs, consider a stock level reporting. Approximately 7000 items are sold daily; for every item sold, a description card is produced, an exception listing of all items below a specific amount is to be created daily, and a total listing of all types and amounts of inventory items is to be prepared monthly.)

The general specifications give each vendor a chance to build a system which makes optimum use of the features of his system. Each vendor is free to use to the utmost any experience and ability he has to prepare the proposal. General specifications thus make maximum use of the vendor's system analysts and permit him complete freedom to produce the best possible system for the user. Rather than relying on the limited experience of the company's two or three analysts, the problems are tackled by the vendor's top analysts, who are more adequately geared for such work.

Innovations to the system may well be the result of such an exercise. Or, a smaller, less expensive system than was thought possible might be proposed as a result of some exceptionally good system work by a vendor. The vendor may propose that some of his package application programs could satisfy many of the system requirements. With software costs equaling hardware costs on today's systems, the resultant cost savings could be important to both vendor and user.

From a systems viewpoint, the user has much to gain by relying on general specifications to describe the applications and needs. However, at the same time many problems are created. With a set of general specifications, a user should expect to spend many hours with vendors and their representatives, discussing possible systems approaches. Countless hours will also be spent trying to verify the systems proposed by the vendors in order to ensure that they will be able to handle the required applications. It is also important that the system concepts proposed are thoroughly understood by the user. Whichever system is selected, the vendor's representative will not be delivered with the system. It will be the user's responsibility to turn the concept into reality.

General specifications also prove awkward and difficult as standards by which to compare competitive proposals and to select a winner. With general specifications, system rewards can be great, in terms of improvements to the computer system, but the difficulties of evaluation can also be great [Yearsley 1973, Chorafas 1967, Tatham 1969, Joslin 1977].

B. DETAILED SPECIFICATIONS

Detailed or specific, specifications are just what their name implies; each and every step to be taken in each of the applications is spelled out. Usually, the synthesis that was used in developing the flow charts for cost determination during the system study is repeated step by step in the specifications. Detailed specifications must be written very carefully to ensure that they do not become machine-oriented rather than application-oriented. Machine-oriented specifications might discriminate against some vendors and thus unintentionally deprive the company of the best system. Since detailed specifications require the vendors to configure their systems exactly as specified, the systems design work for the vendors is simplified, but allows them little freedom to fit the applications to their computers. The computers must be fitted to the applications.

Since detailed specifications are completely descriptive and are fully and uniformly defined to all vendors, they have a definite advantage for the user. Thus, there should be little time wasted in talking with vendors. No system will be proposed that is far inferior to the system represented by the specifications. The submitted proposals may be more easily compared, verified, and evaluated since the proposed systems must all be identical, matching the steps set forth in the specifications. With detailed specifications,

the systems proposed can be no better than the system specified, although the trouble involved in obtaining it should be minimized.

C. OTHER APPROACHES

Specifications do not have to be either general or detailed; they may be at any level in the general-detailed range, in which a certain amount of synthesis may be done by the user (and the user specifies this part in detail), and a certain amount may be left to the imagination of the vendor. Such a method may be called a modified-detailed specification. Its relative merits depend upon a rule of direct proportions: the more general the specifications, the greater the chances of obtaining a superior system; the greater the degree of detail in the specification, the easier the proposals will be to handle. The user can set the level of modification of a full-detail specification by the degree of synthesis he gives to the vendor.

The combination of general and detailed specifications ought to be used in preparing system specifications. In this method, the general specifications are given as the guidelines to be followed in preparing the proposal, and the detailed specifications are given as an example of how the applications might be handled. The use of the detailed specification as an example serves a threefold purpose:

1. It clearly indicates the activities and functions to be performed in each of the applications, and answers many questions that the vendor might otherwise have to ask.

2. It becomes a common starting point for all vendors. The example may be modified differently by the contending vendors but they are all departing from the same basis. It also gives an indication of the level of sophistication being sought in the proposals.

3. It gives the small vendor something good on which to base a bid without necessitating a full system analysis effort.

Preparation of detailed specifications forces the user into the kind of thinking that the vendors will have to engage in. Since the user will be doing the thinking first, he should discover any problem areas before the specifications are released to vendors. This naturally tends to make for smoother relations between the vendors and the user. Proposals submitted in response to this combination specification should all present solutions as good as the detailed specification approach [Tatham 1969, Yearsley 1973, Joslin 1977, Chorafas 1967].

Usually, the type of application to be handled by the computer system indicates the type of specifications to be used.

IV. LIMITATIONS IMPOSED ON SELECTION

The limitations to be imposed on the computer system selected should have been uncovered in the system study. There are two kinds of limitations: mandatory and desirable. The distinction between these two categories should be that the items listed as mandatory requirements are those items that are essential to the implementation of the company's needs.

A. MANDATORY REQUIREMENTS

Mandatory requirements should be stipulated by the specifications, in order to protect the user from considerations of proposals which will not satisfy basic needs. By definition, a proposal will receive no consideration if it fails to meet any one of the mandatory requirements. The less stringent the mandatory requirements, the more likely it is that any given manufacturer will be able to compete in the procurement. Some examples of such mandatory requirements are shown in Figure 1.

B. DESIRABLE VERSUS MANDATORY REQUIREMENTS

The desirable features are only those items which would make the completion of the company's mission easier. Upon submission of a proposal by a vendor, and in consideration of the limitations of both categories designated by the user in his solicitation of proposals, the failure of the vendor's

FIGURE 1

EXAMPLES OF MANDATORY REQUIREMENTS

Minimum turnaround time

Total daily processing time

Compiler availability and requirements

Operating system requirements

Compatibility

Expansion requirements

Translation requirements

Site constraints

Hardware constraints

Equipment qualification

Special software

proposal to possess some desirable feature as designed should invoke some penalty upon the proposal, although it would continue to be considered in the process of selecting the most advantageous proposal.

Computer acquisitions have one thing in common: in most cases, the company wants the best system it can find for the lowest possible cost. In order to stay with the lowest possible cost, most users are interested in general-purpose computers. Special-purpose computers can be expected to run anywhere from 50% to 700% or more in cost than general-purpose ones [Rubin 1971]. The very nature of a general-purpose computer implies certain features that are not readily changeable. The general-purpose computer must be taken largely as it comes from the plant. Thus the user is put into a position analogous to one who has spent years drawing up blueprints for a new house, but who suddenly finds himself with an immediate need for a house. He can have the house built to his blueprints, which will prove very expensive due to unique building cost and the cost of temporary housing, or he can look for an existing house that fills his needs. If in searching for an already built house he is looking for one that exactly matches his blueprints, he may well have to go without a house. But if he is willing to not adhere strictly to the specifications, looks for houses with similar room layouts and other features, and settles upon the one most closely matching his blueprint, then he will have a house, which is his major requirement [Chorafas 1967, Joslin 1977, Tatham 1969].

C. LIMITING CONDITIONS

Now that the dangers inherent in mandatory requirements have been discussed, the several classes of limiting conditions can be described.

1. Cost of the System

A primary consideration in the study of any computer system is the amount of money that the company is willing to spend. Mandatory cost limitations may produce an effect opposite to the one desired. Costs should be reserved for use as evaluating factors and not as limiting factors. If a truly mandatory condition exists, such as that a fund of so many dollars, with provision against its increase, has been set aside for procurement of equipment, then the condition should be stated. Absolute funding of this sort is unlikely to exist. System costs normally should be handled as a desirable limitation.

2. Due Dates

One of the first things encountered in a system study will be the existence of due dates. As a matter of fact, these dates are not often really mandatory and they should not be treated as mandatory requirements but as desirable ones to express wishes and not commands.

3. Application Capabilities

The purpose of the study is to get a system capable of handling the applications specified. However, the possibility remains that some of the desired capabilities are not available. It is better to handle application requirements as desirable limitations.

4. Responsiveness

This is a limitation factor akin to application requirements. Responsiveness requirements should be regarded as desirable limitations, as are most application capabilities. Since responsiveness is the element of greatest interest to the top executives of the company, it can easily be made absolute by the statements of the top executives.

5. Compatibility

The compatibility of the old system with the new should definitely be considered, but whether it should be mandatory is questionable. There is always temptation to make compatibility with the present system a mandatory requirement of the new system.

6. Vendor Support

Often seen in specifications, and stated as mandatory requirements, are conditions which relate to the type of support that must be available from the vendor whose proposal is accepted. They can be mandatory or desirable limitations depending on the particular cases of applications.

7. Reliability

Reliability is a condition inserted into most specifications as a mandatory limitation, generally expressed in such terms as "the system must have 95% uptime". Specification should state that reliability will be a factor in evaluation and selection of a proposal, however, the value of reliability should not be exaggerated or expressed in such absolute terms as to prevent the exercise of judgment.

If reliability is to be used as a mandatory requirement, then some means of measuring it will have to be determined. Usually the problem is that the measurement of the reliability would take more time than could be allowed. Or, as in the case of real time systems where reliability really is a mandatory limitation, complete redundancy of the system may be required.

8. Space Requirements

The dimensions of systems may change from selection to selection. Space requirements is a desirable limitation. In most cases the immediately available space could be extended or new space found. A proposal should not have to be discarded just because it requires more than the allotted space, especially when the allotment might have been determined arbitrarily or thoughtlessly.

9. Input/Output Requirements

Input/output requirements are expressed in terms of forms or formats to be used, number of copies to be prepared, and the like. Input/output requirements which are limiting conditions should be restudied and reappraised. They should be viewed in two ways:

a. What will happen to the system if these requirements are changed?

b. If these requirements remain unchanged, can vendors meet them?

10. <u>Other Limitations</u>

Many other limitations may appear in a specification. The deciding factor in establishing them as mandatory requirements is whether each item is important enough to warrant throwing out the proposal completely if it fails to meet the limitation to any degree. This happens rarely, but it can happen [Chorafas 1967, Tatham 1969, Joslin 1977, Wooldrige 1973].

V. PROPOSAL EVALUATION TECHNIQUES

A computer evaluation methodology is simply a planned method for selecting the most satisfactory computer system from a number of satisfactory computer systems. The evaluation methodology tries to assure that all of the computer systems in the final phase of selection are satisfactory; i.e., that they meet all the basic requirements of the solicitation and that they are what the vendors represent them to be. When thought of in this sense, evaluation may sound like a rather trivial task, since any resultant selection will, by definition, be satisfactory. However, there are different degrees of satisfaction and different groups of people to be satisfied.

In evaluation, higher levels of satisfaction and satisfaction of other than the basic requirements are considered. An important group that must be satisfied with the evaluation process is the vendors. The vendors will not spend the time and money necessary to bid the "satisfactory systems" that get evaluated, unless an evaluation methodology has the appearance of being fair and unbiased. If they have bid but feel they have not been fairly evaluated, they will protest the selection.

Within the government, or any large organization, a protest can lead to considerable embarrassment for the procuring activity if upheld and, in any event, will consume

a great deal of time and effort in resolving the protest. Therefore, a selection methodology must be satisfactory to the vendors as well as to the procuring activity [Auerbach 1975, Clifton 1969].

The organization of the evaluation must be carefully structured so that the participants are aware of their individual areas of responsibility. A hierarchical arrangement is necessary in order to have increasing levels of responsibility as the decision areas become broader.

In order to select the best computer system after specifications have been determined, the following need to be considered:

* possible computer systems;
* by whom the selection is to be made;
* selection methods;
* the criteria used in the selection process and their relative importance.

The information to be used in selection methods can be obtained from:

* published surveys and reports,
* service and product publicity material,
* hardware, operating system and program documentation,
* managerial, sales and technical staff,
* in-house staff and other users of the machine or service [Webster and Johnson 1977].

To give an idea about competitive selection time, succeeding steps in the competitive selection process for

the Navy are outlined in Table 1, which shows the estimated time to accomplish each step. The complexity of the system being acquired generally determines the length of time required at each step [Prokop 1976].

An evaluation methodology should:

* consider those items or features wanted but not mandatory,
* cover all the items or features desired,
* facilitate the establishment of meaningful and understandable relative values between all the desired items,
* require the completion of the previous criteria before the solicitation document is completed,
* permit disclosure of all desired items and their relative values to the vendors,
* incorporate systems life costing.

An evaluation methodology that satisfies:

* all the listed criteria is a SUPERIOR methodology,
* five of the listed criteria should be considered a GOOD methodology; but before settling for it, a superior methodology should be sought,
* only three or four of the listed criteria may be considered a FAIR methodology, but it should be possible to find a better methodology,
* less than three of the listed criteria would have to be considered a POOR evaluation methodology and should not be used [Auerbach 1975, House 1976, Chorafas 1967].

TABLE 1

TYPICAL COMPETITIVE SELECTION TIME FRAME

Draft request for proposals for approved project	30-90 Days
Release of draft for comments	30 Days
Revision of request for proposals	30 Days
Response to request for proposals	60-120 Days
Evaluation of proposals and benchmark	30-120 Days
Administrative time after evaluation	20-60 Days
Installation of equipment after contract award	90-270 Days

Range: 290-720 Days or 10-24 Months

Prokop 1976

A. TECHNIQUES FOR THE EVALUATION OF COMPUTER SYSTEMS

There are several techniques for the evaluation of computer systems. They generally fall into one of two categories. Either they are very simplistic in that they tend to ignore most of the criteria listed previously, or they are sophisticated and incorporate most of those criteria [Auerbach 1975].

1. Simple Techniques

Simplistic methodologies are better known but less successful techniques. Theoretically, they are not worth much discussion, but they illustrate the need for the more sophisticated techniques.

a. Sole Source

"I have been happy with this vendor. Why should I change?" It is possible that one might be happier, for less money, with another vendor.

b. Subjective Judgment (Overall Impression)

Probably the most frequently used evaluation approach is to have no preestablished approach, just some general statement such as: "When the proposals come in, an unbiased group of evaluators will look through them and pick the one that provides the most benefits at the lowest prices." People who advocate this approach to computer selection will ridicule any attempt by a prospective user to preestablish which items he must consider in an evaluation before he receives the proposals to be evaluated. They will ridicule the prospective user even more if he attempts to establish specific values for each of these items.

Prospective evaluators will argue that until the prospective user has received all the proposals for computer systems, he will not know all the items. They will point out, for example, that if all the vendors propose any given major item, then its importance to the selection process is negligible; whereas even a minor item can have significant influence on the final decision if it is proposed by one or two of the vendors but not by all. These prospective evaluators want to make their selection first and then justify their evaluation.

This procedure is a comfortable one for the evaluator since he is not forced into doing any advanced planning. Almost any vendor who meets the mandatory requirements could be selected as the winner under those circumstances. All it takes is an evaluator who is clever with words and who can accentuate the strong points of the winner and flaunt the weaknesses of the losers. However, it is unfair to the vendors and, in the long run, also unfair to the prospective user to establish the criteria for evaluation after the proposals are received. This kind of evaluation is subject to the vagaries of human nature, over which there is no control [Joslin 1977, Rubin 1971].

c. Cost Only

This technique advocates selecting the lowest cost system that meets all of the mandatory requirements. However, what if the next-to-the-cheapest system is only slightly more expensive than the cheapest one, and yet would

far outperform it? Due to the unanswered cost and requirement questions, the cost-only approach is rapidly losing favor, except for smaller systems with static workloads [Joslin 1977].

Any meaningful evaluation methodology should differentiate between mandatory and desirable features. Either a vendor shows that he can meet all the mandatory requirements or his proposal is not considered for evaluation. If only one vendor can satisfy all of the mandatory requirements, he is automatically the selected vendor.

Suppose, however, that three vendors were to satisfy the mandatory requirements, then the proposals of all three would be considered to be equally satisfactory. The purpose of the evaluation methodology is to establish some logical and defensible means of differentiating between the proposals of these satisfactory vendors and selecting the one that is best suited to meet the activity's needs. The items used to differentiate between the vendors who have satisfied the mandatory requirements are desirable requirements. Since there are many desirable features of varying importance, the evaluation methodology must find some method of establishing the relative values of these features and their relationship to the system cost.

d. A Case History

Company ABC wanted a computer system. The company needed to have a given set of problems processed within one hour's time, and there were certain other

requirements that had to be met. ABC sent out a request for proposal, to which five vendors responded. Three proposals satisfied all requirements. The only significant difference between the three proposals was the amount of time it would take to process the problems and cost of the three different systems. The findings were:

Vendor	System Cost ($)	Time to Complete Problems (Min)
X	300,000	50
Y	275,000	55
Z	500,000	25

Vendor Z was selected by the evaluators because they interpreted the time to process the problems in its reciprocal sense of how many sets of problems could be processed per hour. Thus, vendor X could process 1.2 sets; vendor Y could process 1.1 sets; and vendor Z could process 2.4 sets. They then divided each system's cost by the number of sets which the system could process, with the following results:

Vendor	Cost Per Set Per Hour
X	$250,000
Y	$260,000
Z	$208,333

The evaluators justified their choice by pointing out that vendor Z's system gave the most room for expansion and could process more sets of problems per dollar than either of the other systems [Auerbach 1975].

2. Sophisticated Techniques

There are two basic evaluation methodologies which permit the evaluator to consider desirable features and establish the relative value of the desirable items. The approaches are weighted-scoring schemes and cost-value based approaches.

a. Weighted-Scoring Technique

Under this system, the prospective user preassigns varying quantities of points to all items he considers important and then selects the system earning the most points. An example of this technique is shown in Table 2. Vendor B would be selected [Auerbach 1975].

Since this technique appears to satisfy all of the criteria listed, it may seem to be a good evaluation methodology. However, upon closer review it is found to fall down on the criterion which calls for "meaningful and understandable" relative values between all desired items and on the other criterion which calls for incorporation of system's life costing. (See page 30 for the criterions.)

It is difficult to establish a meaningful and understandable relationship between the number of points awarded for low cost. For instance, the example shows cost having a weight of 70%, but why 70 rather than 30 or 50 or 90%? The failing of this technique is that there is no common denominator among the items being weighted. Thus, there is no "meaningful and understandable" relationship.

TABLE 2

AN EVALUATION USING WEIGHTED SCORING

Evaluation Items	Relative Values %	Vendors' Scores		
		A	B	C
Cost	70	70	60	50
System potential	20	7	16	20
Technical characteristics	5	2	4	5
Vendor support	5	3	4	5
TOTAL SCORE	100%	82	84	80

Until a meaningful approach can be found to the proper distribution of points between the items desired, the weighted-scoring technique will never be considered a very satisfactory evaluation technique.

b. Cost/Effectiveness Ratio

This technique is simply a subcategory of the weighted-scoring technique, except that with it, by dividing the total system cost by the sum of the points scored in the other desirable categories (effectiveness category), the prospective user can select the system with the lowest ratio of cost to effectiveness. However, such a division of points is generally not sufficient to establish a meaningful relationship between cost and effectiveness [Joslin 1977, Borovits 1975].

c. Cost-Value Technique

None of the previous evaluation techniques proved very satisfactory under intensive investigation. Therefore, a new evaluation method, the cost-value technique, was developed in 1964. This technique combined the simplicity of the cost-only technique with the realism of the weighted-scoring technique. The result was a technique superior to both. It is superior to the cost-only technique because it considers some items in a computer system to be of value in addition to the system's cost and its compliance with the mandatory requirements; and it is superior to the weighted-scoring technique in that it establishes a meaningful

relationship between the items of value and the system's cost while at the same time incorporating system's life costing [Chorafas 1967, Joslin 1977].

The cost-value technique recognizes the necessity of evaluating the desirable features offered by the various computer systems proposed. With this method, the desirable features and the cost associated with the system are all that are evaluated; that is, the ability of the proposed systems to perform the functions for which a computer is to be procured and the ability of the vendors to meet any other conditions specified as mandatory in the specification package are not evaluated, but are validated. If it is found that the vendor or his system cannot perform as required, the proposal is eliminated from further consideration. With this technique a company can study any extra features offered in the proposals to determine whether the claimed extra features are important in themselves or are mere incidental elements that appear to be extra features. For example, a 60-nanosecond memory and a 10,000-card-a-minute card reader are not important features in themselves. More important and desirable is the amount of slack time that exists in the proposed system on account of these high-speed units. A study should be initiated to determine the value of every extra feature which is considered to be important.

A distinguishing feature of the cost-value technique is the assignment of the value associated with the desirable features in terms of cost, that is, dollars, of

value. By assigning a cost value, in dollars, to the desirable features offered by the various vendors, a common denominator is provided by which all offered desirable features may be related to each other and to the system's cost. Although the cost values assigned to the various desirable features still will be a matter of each individual selection, and will continue to reflect the opinions of the assignors, a value, when assigned, can be understood, examined, discussed, and changed independently of all other individual assigned values.

An important benefit derived from the use of assignment by cost value is that management can understand what is going into an evaluation, and is able to make informed decisions on the value of any disputed desirable features. The specific cost values established for each of the various desirable features found within each of the proposals are then used for the scoring of the proposals. In the cost-value technique, the proposals are scored or ranked by what will be referred to as a cost-value accounting scheme. This is cost and value accounting, since some of the values and costs used, although stated in dollar terms, may not involve real expenditures.

The cost-value technique amounts to taking the total cost of a system proposed and then deducting the cost values of all the desired extras included in that proposal. The difference represents the derived cost of satisfying the mandatory requirements stated in the specification package.

The system having the lowest derived cost for satisfying the mandatory requirements becomes the system selected, since the values of desirable features offered already have been taken into consideration in deriving this cost for satisfying the mandatory requirements. This ranking also can be looked at from a value-to-cost ratio, but the results will be the same if value is considered in its full sense, as value of mandatory requirements plus the value of the desirable features offered, and cost is considered to be the total cost of the package over the estimated life of the system [Joslin 1977, Tatham 1969, Auerbach 1975].

(1) Using The Cost-Value Technique. The cost-value technique's approach to the extra features (those proposed features above and beyond the mandatory requirements likely to be offered by the vendors) is to appraise them to determine whether they are worthy of inclusion in the evaluation, and, if so, to determine the cost value of these features. To avoid any bias or appearance of bias on the part of the evaluators, and in order to be fair to both vendors and the user, this study preferably should be initiated before the proposals are received. However, it should be noted that the cost-value technique actually is open-ended; that is, if any unexpected extra features are offered, they can be included as part of the selection, if deemed important enough, by simply assigning them their cost value. It thus becomes necessary to deal with either hypothetical or realistically anticipated features. A sample listing of

features which may be considered in cases must first be established; then it can be seen how to go about assigning cost values to those items which really have value to a selection. Table 3 contains a sample listing of these features.

COST ITEMS. All cost items must be considered in the evaluation. Items such as the cost of supplies or personnel may prove to be nondifferentiating in a given selection, but they should not be deleted from the evaluation list because, differentiating or nondifferentiating, they are still true costs associated with completing the specific applications.

Treating cost items as one-time costs or continuing costs is a matter of cataloging. The following rules must govern any proper treatment of cost items [Joslin 1977]:

1. The costs must be spread proportionately over the expected life of the system.

2. The system costs must change to reflect the costs of any planned system expansion.

For example, if the life of a system is set at six years and a uniform expansion rate of ten percentage per year is expected over the life of the system, then each of the continuing cost items on the list, if applicable to the yearly system cost, should be charged for six years. Thus it would be expected that the equipment cost for the sixth year would be larger than that for the second year.

TABLE 3-a

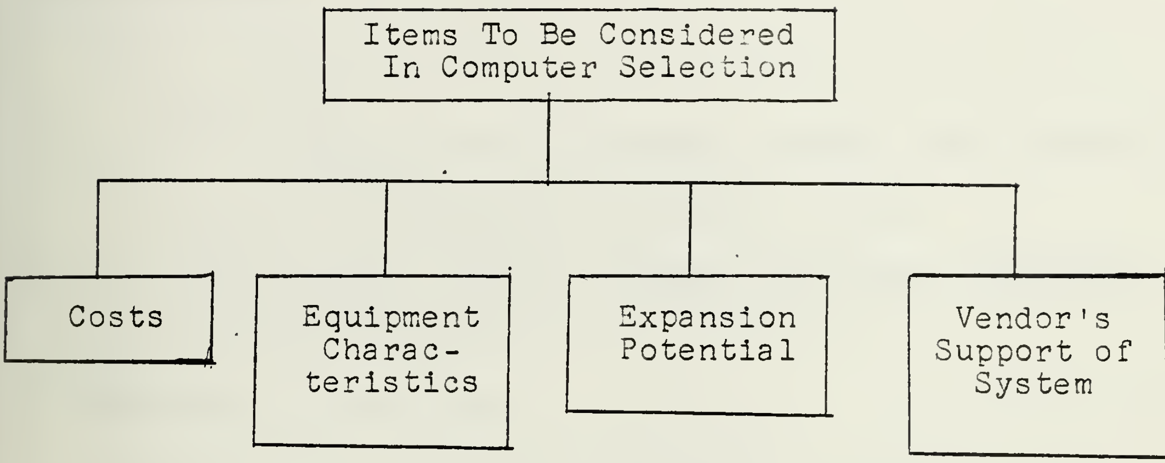

Items To Be Considered In Computer Selection
Costs
Equipment Charac- teristics
Expansion Potential
Vendor's Support of System

TABLE 3-b

COSTS

ONE-TIME COSTS

Site Preparation

- Electrical
  - Air conditioning (cooling, heating, and humidity control)
  - Power supply (including all wiring)
- Space for equipment
  - Facilities (walls, ceiling, painting, draperies)
  - False flooring (including bracings)
- Security provisions

Equipment Installation

Equipment Transportation (including insurance cost)

Vendor Support

- Personnel (analysts, programmers, operators, instructors)
- Training (including transportation, living costs)
- Existing programs
- Backup facilities
- Machine time (checkout)
- Documentation
- Program and data conversion

CONTINUING COSTS

Procurement of Computer System Equipment (falls in one-time costs category if system is purchased)

- Central processor and associated equipment (console, floating point option, real time option, etc.)
- Peripheral computer equipment: on-line or off-line (remote-inquiry device, card reader, printer, etc.)
- Auxiliary equipment
  - Keypunch machines and other data-created devices (flexiwriter, teletype machine, etc.)
  - Printers, sorters, collators, etc.

Operation and Maintenance of All Electrical Equipment

Personnel (manager, analysts, programmers, operators, etc.)

Program Development

Supplies (magnetic tape, printer paper, cards, etc.)

Indirect Cost for Space Used

TABLE 3-c

EQUIPMENT CHARACTERISTICS

SPEED

Time required to complete applications specified

Instructions
- Add time (fixed and floating)
- Mult. time (fixed and floating)
- Divide time (fixed and floating)
- Move
- Other instructions (through all other instructions thought significant)

Peripheral equipment
- Printer (lines per minute)
- Card reader (cards per minute)
- Card punch (cards per minute)
- Magnetic tape units (characters per second)
- IAS (characters per second, average)
- Other equipment (through all other peripheral equipment listed)

CAPACITY

Storage capacity of main memory (core)
Storage capacity of immediate-access storage (IAS)
Storage capacity of magnetic tape
Characters per printed line

COMPATIBILITY

Program; tapes; cards

RELIABILITY

Error detection; error correction techniques; mean time to failure, etc.; redundant components

SPECIAL FEATURES

Memory lockout; parallel processing

PROBLEM TIMINGS

Central processor limited; input/output limited

SWITCHABILITY

Magnetic tape units; printers

OTHER CHARACTERISTICS

Size of equipment (each piece considered); weight of equipment (each piece considered)

TABLE 3-d

EXPANSION POTENTIAL

SLACK TIME (amount of available free time on each piece of equipment)

Central processor, magnetic tapes, immediate access storage, card punch, printer, remote terminals, etc. (through all other equipment offered)

MAXIMUM EXPANSION (number of units that can be added to system)

Magnetic tapes, immediate access storage, card punch, printer, etc. (through all other system equipment offered), extra core, disk drives

COMPATIBLE EQUIPMENT

Larger processors

Higher performance units

TABLE 3-e

VENDOR'S SUPPORT OF SYSTEM

PROGRAM ASSISTANCE

Development; writing; converting; emulating

TRAINING

Analysts; programmers; operators; managers; users

MAINTENANCE OFFERED

BACKUP AVAILABILITY

PROGRAM TESTING

Hours; shift; location

EXISTING SOFTWARE

Operating system

Schedulers; input/output control; memory allocation; etc.

Sort; merge; system simulators or emulators; COBOL; FORTRAN; report generator; etc.

DOCUMENTATION

PERSONNEL LOANED

Analysts; programmers; operators; users

Another important consideration relating to cost items is that they should show the cost for individual pieces of equipment to be used. This should be true in all cases except when the system is to be used for less than one shift, or when the entire system is to be purchased.

No cost items should be duplicative; that is, the system should not be charged twice for the same equipment or service. For example, if a card reader is used both online and offline, its full cost should not be shown twice. Similarly, program development, if performed by the user's personnel rather than the contractor's personnel, and personnel cost should not both be costed for this program.

EQUIPMENT CHARACTERISTICS. The significance of the characteristics of any piece of equipment is measured in terms of the running time of the system, which in turn determines the system's cost and expansion potential or its system responsiveness. Typical of the kind of equipment characteristics now being discussed are: the relative speeds and capacities, hardware compatibility, switchability, reliability, and special features. For real time systems, these conditions usually will be stated as mandatory requirements [Chorafas 1967, Coutinho 1977].

Sample problem timing items (times required to perform the benchmark problems) should not be evaluated. They should be used exclusively for validation or establishment of application timings quoted in the proposals. This

application timing, in turn, is the base on which system cost and expansion potential are calculated. Since its value is felt in the other items, it should not be evaluated.

EXPANSION POTENTIAL. The expansion potential of a system is considered to be an important extra, since it allows for growth beyond the specified amount. Thus the system has the possibility of a longer life and of handling larger workload peaks. Another type of expansion which is sometimes important is the ability to add on different types of peripherals.

VENDOR SUPPORT. Vendor's support is also deemed important to the cost-value technique. All of the items of vendor's support could be desirable features, since each offer could result in some actual saving to the user [Thrussell 1976, Sabol 1972].

(2) Constructing Evaluation Templates. The cost-value technique examines expansion potential by evaluating:

1. The system's ability to handle additional workloads, and

2. The system's ability to handle different peripherals.

To evaluate the system's capability for handling additional workloads, it is necessary to first calculate the run time required by the system to complete all required applications. The elements and aspects of a computer and its use that must be considered in any calculation of the running time of a system are:

1. Speed
   a) Central processor
   b) Peripheral equipment
   c) Auxiliary equipment
2. Capacity
   a) Central processor
   b) Peripheral equipment
3. Special features
   a) Parallel processing
   b) Simultaneous operations
   c) Other
4. Software efficiency
   a) Compiled languages
   b) Assembled languages
5. Reliability
   a) Switchability
   b) Error detection and correction features
6. Preparation time
   a) Setup/take-down
   b) Program insertion
   c) Media handling
7. Nonproductive time
   a) Reruns
   b) Program checkout

If a system were used 24 hours a day for 30 days each month, it would be possible to get 720 hours a month of computer time. However, most manufacturers require

about $1\frac{1}{2}$ hours a day for preventive maintenance of the machine and this would leave about 675 hours a month. From this 675 hours must be subtracted various loss factors, such as unscheduled maintenance, idle time, setup time, machine malfunction time loss, program housekeeping, development and maintenance of programs, program errors, and operator errors. These reductions cut the actual maximum time available for production to between 400 and 500 hours per month for most business applications and 500 to 600 hours per month for scientific applications [Joslin 1977, Webster and Johnson 1976].

The figures for total production time available should be used to deduct the times computed to process the monthly workload. The time remaining, called slack time, is the time available for expansion. The amount of slack time available could be increased by reducing the time required to process the monthly workload, which could be achieved by adding processors of higher capability. However, more or faster units should be added only when the value of the additional slack time is greater than the cost of modifying the system to make this additional time available. The worth of the additional slack time might be considered as the additional system life brought about by the expansion potential. The concept of system life applies not only to a purchased system, but also to a leased system where there is extensive investment in software and know-how which is geared to the existing system. Any changeover thus might prove costly.

YEARLY EXPANSION. The most meaningful way of preparing a value template for yearly expansion would be to look at the stated workload for each year, estimate the confidence level that the stated workload is correct, then increase the workload until a confidence level of about 95% is obtained. For example, if it were estimated that the stated workload for the first year was 100% of some base amount, after considering the case it might be found that there was only about an 80% confidence in that estimate. However, if that base amount were increased to 110% of the old base (having a confidence level of 80%), there would be 88% confidence; if it were increased to 120% of the old base, there would be 96% confidence. The estimate would have to be increased to 125% of the old base before there would be 100% confidence that the workload as then stated could not be exceeded in the first year [Joslin 1977].

Assume that the system envisioned for this case was expected to lease for $100,000 a year. With these facts, the following value template might be established for the value of expansion for the first year:

First Year Expansion Value Template

Confidence Level Percentage Expansion	Value
75	$20,000
20	$16,000
10	$ 8,000

The value figures are derived by saying that, if the user were willing to pay $100,000 to handle what he is only 80% confident represents the first year's workload, he should be willing to pay 25% more to have 100% confidence in the system's ability to handle all the first year's workload; that is a total of $120,000, or an increased value of $20,000.

Similarly, to be 96% confident rather than 80% he ought to be willing to pay 20% more, or $16,000, and so forth. In a similar way, an evaluation template could be prepared for each of the years.

If the evaluation templates are supplied to the vendors, there should not be any need to adjust the vendors' proposals to reflect the greatest value for the user. If the vendors are not supplied with the evaluation templates, then the value of expansion potential must be calculated for each year. For example, if a vendor were to propose a system that was so modulated that every year his system took all the time available just to handle the required workload, but examination of his equipment revealed that he could increase his system's capability by 10% for a yearly lease increase of $2,000, or 20% for an increase of $6,000, or 75% for an increase of $13,000, adjustments to his value of expansion potential should be made.

If the previously established Value Template were to be used, the vendor's yearly values for expansion would be:

Increased Value Ratio To	Increased Lease Price
$ 8,000	$ 2,000
$16,000	$ 6,000
$20,000	$13,000

The best difference of value minus cost is $16,000 - $6,000 = $10,000; a 20% confidence expansion is indicated.

A slightly different approach for determining the relative value of yearly expansion could make use of marginal utility analysis techniques, but similar results should be obtained [Webster and Johnson 1976, Thrussell 1976, Joslin 1977].

EXPANSION BY NEW OR DIFFERENT PERIPHERALS. There are times when it is of definite value to be able to add peripherals to the system that were not called for in the basic system requirements. For example, it might be possible to handle a given application without using immediate access storage. However, if the user feels that sometime in the future he might wish he had experience with immediate access storage (IAS), he might establish a value for having the system possess the capability of connecting as IAS unit. In fact, he might establish two values: the value of having IAS proposed and a lesser value of having the capability to add an IAS unit.

The suggested method for determining the value of such a capability is:

1. Calculate the probability of needing the capability.
2. Determine the cost of obtaining the capability independent of the present system.
3. Take the product of these two figures.

It must be remembered that there is likely to be considerable difference in the evaluation of the various capabilities proposed, since no one application measures every possible capability.

EXPANSION WITHIN A FAMILY. The advantage of this type of expansion is that the programs written for one of the smaller computers in a family will run on the larger ones. Therefore, the only expansion cost is that of the new computer, not a reprogramming cost. To the extent that this statement is true, the family approach could be used in a fashion similar to the extra system life approach. However, the inefficiencies of running programs on a large computer that were prepared for a smaller one must then be considered.

VENDOR SUPPORT. There are several methods of assigning cost values to vendor support items. The simplest method, for which the user cannot estimate the value of the support items, is to require the vendor to quote costs for various levels of performance. For example, if one vendor offered on-site maintenance while all the rest offered on-call maintenance, a feeling for the maximum value of the on-site maintenance could be ascertained by asking each of the other vendors to state the cost of such service. Sometimes, however, the cost quoted may be so excessive as not to make a fair base against which to award value. For example, if a user were impressed by some special programming routine and asked various vendors for the cost of

supplying it, he might receive answers in hundreds of thousands of dollars, whereas if he himself were to go out and procure such a routine, he probably would not be willing to pay over $5,000. In such a case, the $5,000 should become the base. Restated as a generalized rule: In cases where the user would place a value on a service lower than a vendor's cost, this value figure becomes the base for determining the item's value [Joslin 1977].

In some cases, the vendor may not be able to give cost figures for supplying service equal to some of the levels desired, simply because he does not have the necessary facilities. In other cases, it might be practical only for the vendor himself to provide the service. An example of this is a special training requirement which might occur in a real time system, where some provision, probably a special program, must be provided to allow the trainees access to the remote consoles for training purposes, yet prevent their mistakes from destroying the good system. This kind of training aid probably can be provided only by the given vendor. In such cases, the cost value of such a service must be determined individually, and might be considerably higher than the costs charged by any other vendor. But the higher cost-value figure should become the base.

The cost value of these items also might be ascertained by the user, by taking each item in turn and determining its value to him. Among these cost-value items,

the one most closely representing the user's needs should be chosen. However, the cost value should never exceed the cost of having the service contracted by someone else.

Some items such as available backup and debugging facilities are support items on which the vendors cannot be asked to change or improve. Therefore, their cost value must be evaluated as the items are proposed. An approach to determine the cost value of back-up would be to determine the probability of experiencing a catastrophic failure, then the cost associated with carrying on the computer activities on the back-up facilities available. The cost times the probability of catastrophe should give the probable value for back-up of each of the various systems. Cost-value determination for debugging facilities could be handled in the same way [Chorafas 1967, Sabol 1972].

OTHER DESIRABLE FEATURES. Many other features might be considered in hardware selection. Items such as memory lockout or desirable compatibility can be handled by determining the cost that will be eliminated by the inclusion of such abilities. Thus the costs that would have to be paid to convert tapes of one kind to another would be saved if the two systems were compatible. This cost becomes the cost value.

Being able to run a portion of the old programs on the new system is a desirable feature. Therefore, program compatibility (or portability) is another important aspect of compatibility. An estimate can be made of the cost

that would be incurred if that portion of the software had to be run elsewhere until rewritten. This estimated cost becomes the cost value of program compatibility. However, if the compatibility is achieved through the use of an emulator or simulator and the resultant programs would not run at the efficiency of rewritten programs, then the value of this compatibility is decreased. The amount of the decrease would be dependent upon the frequency of use of the programs. Infrequently used programs do not need to be as efficient as frequently used programs.

Costs of the time and trouble that could be saved by inclusion of a memory lockout device become its cost value, which can be shown in an evaluation template [Auerbach 1975].

A system may be proposed that will enable management to have access to any information within the file in less than one minute, or to have management reports ready by 1:00 p.m. everyday. In such cases, a study must be initiated to determine the cost value to management of being able to have one-minute access, rather than ten-minute access as requested in specifications package, or the cost value of having the reports ready by 1:00 p.m., rather than 3:00 p.m. as similarly requested. Where possible, these value assignments should be made in time to help the vendors with their bidding.

With real time or time-shared systems, another area of desirable features should be considered.

For example, a time-shared system may call for eight remote terminals with an inquiry from any terminal being handled within six seconds. Some of the systems proposed may be able to handle two or three times the required number of terminals. The value of these extra terminals depends upon the probability of their profitable use or on some logic similar to that used in making the original decision that eight were required. The value of various speed responses should be determined and shown in an evaluation template.

Another extra is the possibility of coming across an innovation or a new approach to the system. The prospective user could assign a cost value to such a new approach by making a realistic determination of the savings that are likely to accrue if he uses the suggested approach times the degree of probability that the suggested approach will actually work, or by estimating how much it would have cost him if he had had a special study made that might have come up with the same recommendation.

Extras such as a purchase option offered or expected trade-in will or will not have value, depending upon the type of procurement plan to be used in the acquisition of the system.

(3) General Thoughts On The Cost-Value Technique. In applying the cost-value technique, the following points should be kept in mind:

1. The methods described here for cost-value determinations are by no means the only ones that might be used.

2. The items chosen for discussion are not the only items worthy of inclusion in a cost-value evaluation, nor will they all necessarily appear in any given selection. The circumstances of a specific selection determine the items to be used.

3. There is nothing sacred in any of the cost values established, since the value of any item depends upon the likelihood of the user's need for that item. For instance, if the described system is to be used only for one or two applications and the size and volume of these applications are fixed then the cost value of expansion potential is likely to be nil. On the other hand, if the described system is the first system to be installed in a growing company, the cost value of expansion potential will be high because every hour of available expansion might be as valuable as each hour in actual use. If the described system is to replace an existing, compatible system, some of the vendor support items, such as personnel loaned or program assistance, may have no cost value. However, if the computer is for a relatively inexperienced group, such factors might have a cost value as high as, or higher than, $40,000 per man-year [Joslin 1977].

Calling the items to be evaluated "extras" implies that the "extra" is the amount over the minimum acceptable or mandatory. The value of an extra may be established independent of the proposals from preconceived values

which are created to show value for varying amounts of each of the extras evaluated. These preconceived ideas of worth are referred to as "evaluation templates".

ESTABLISHING LIMITS. To assess the value of any given item is a difficult task. The logical starting place is what the item costs. If the item is competitively available, its value should never greatly exceed its cost. For example, if the cost of having a mathematical subroutine written by a software consulting group is $10,000, then it would be reasonable for a user with little use for this subroutine to establish a value of only $500 for its availability.

If only one vendor can supply a critical subroutine, its value is almost indeterminate. However, this case should not arise in cost-value assessment for two reasons:

1. If the item is critical, it should be listed as mandatory requirement and should not require value assessment.
2. If the item can be procured from only one vendor, the full selection ought to have been handled as a sole-source procurement, again making value assessment unnecessary.

DIMINISHING VALUES. The prevailing thought behind most of the evaluation templates created is that, as more of an item becomes available, the worth of that item decreases. This can be shown mathematically by exponential curves such as those shown in Figure 2. However, for ease of

understanding, it is usually easier to break down each value assignment into a group of smaller approximations [Chorafas 1967, Tatham 1969, Sabol 1972, Joslin 1972].

Suppose that the ability to be able to expand by 20% is worth \$20,000; by 40%, \$32,000; by 60%, \$40,000; by 80%, \$46,000; and by anything over 100%, \$50,000. An exponential curve could be fitted through these points and the curve established, but it is usually not worth the effort. Using normal interpolation techniques between the defined points, the value of any expansion capability can be found. Thus, the value of having 50% expansion capability could be found by taking:

$$\frac{\text{Difference between 40\% and 50\%}}{\text{Difference between 40\% and 60\%}} \qquad \frac{10\%}{20\%} = \frac{x}{\$8000} \qquad \frac{\text{Difference in value for 40\% and 50\%}}{\text{Difference in value for 40\% and 60\%}}$$

The unknown difference is found to be \$4,000. When this amount is added to the amount for 40%, the resultant value for 50% is found to be \$36,000. Figure 3 shows this template plotted, using straight-line extrapolation between the defined points and using an exponential curve. The value for 50% expansion, if taken from the exponential curve, would be approximately \$37,000.

An explanation has been given of some techniques for determining the cost values of a number of items that should be included in any selection. The cost values derived for the various vendors are applied as credits to

FIGURE 2

True Exponential Shape of Normal Evaluation Templates

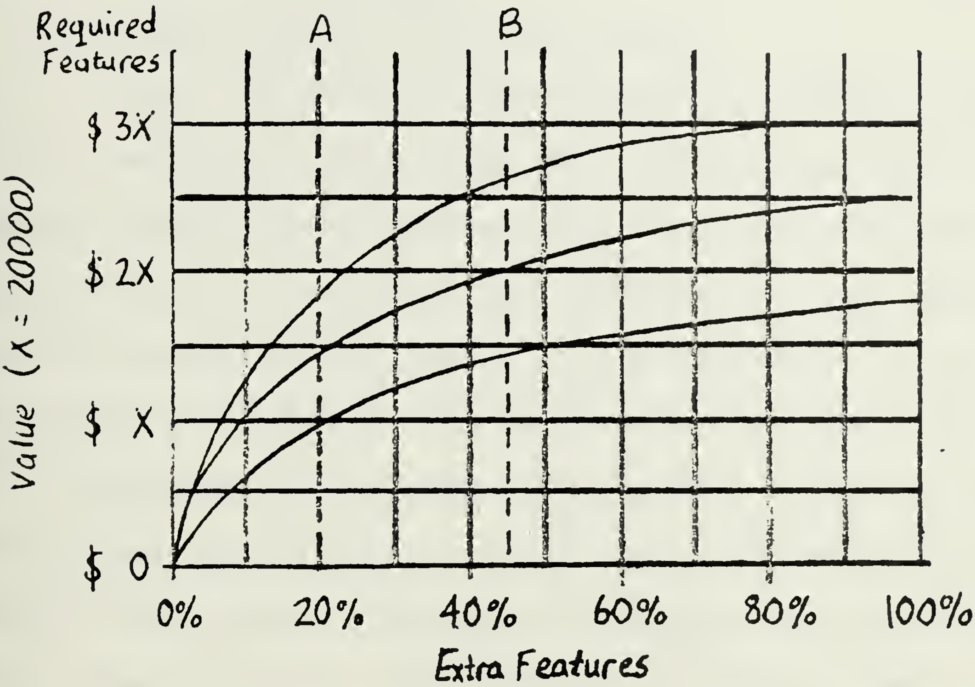
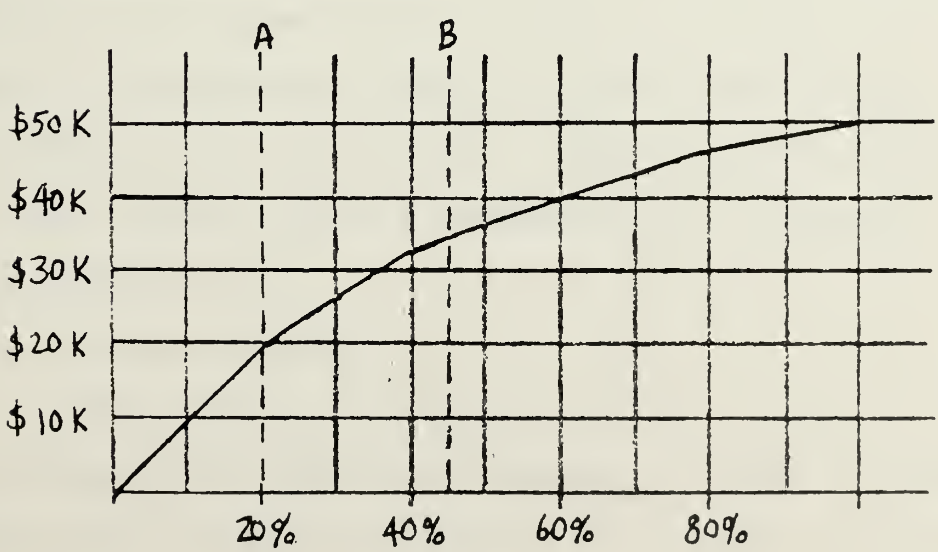


FIGURE 3

Evaluation Templates



offset the costs of the system and the proposed services. The vendor whose proposal shows the smallest difference in out-of-pocket costs minus credits is the one to whom the contract should be awarded.

d. Requirements-Costing Technique

This technique is conceptually the same as the cost-value technique, only under this approach a vendor is assessed a preestablished dollar value or worth for any desirable feature not offered (or offered at a cost that exceeds its worth) by the vendor in its proposal; or if the vendor offers the feature, but at some charge, then the vendor is assessed that charge. The system selected is the one having the lowest overall total cost (including not only the cost of the vendor's hardware, software, and services, but also other costs for such items as staffing, power, air conditioning, etc., and assessments for features not offered). An example of requirements costing is shown in Table 4.

The requirements-costing technique and the cost-value technique are essentially identical, and they prove exceptionally satisfactory once the dollar values of the desirable features are established. Both of these techniques meet all the criteria listed as essential for a superior evaluation methodology.

e. Dynamic Approach

The problem and an approach to a solution is presented here in the form of an example. Assume an organization which has decided to replace its existing computer

TABLE 4

AN EVALUATION USING REQUIREMENTS COSTING TECHNIQUE

Evaluation Items	Maximum Values $	Vendors' Cost $ A	B	C
Vendor Costs		1,000,000	1,200,000	1,300,000
Other Costs		100,000	95,000	90,000
Assessments				
System Potential	400,000	240,000	100,000	20,000
Technical Characteristics	200,000	60,000	35,000	10,000
Vendor Support	50,000	40,000	25,000	0
Total Cost	-------	1,440,000	1,455,000	1,420,000

Auerbach 1975

system in preparation for a major expansion in its information processing activity, proposals are on hand from three manufacturers, A, B, and C, each of whom has a number of systems to offer, named A1, A2,...C4 and benchmarks have been performed for a representative sample of the organization's workload on one or more systems of each manufacturer.

The least powerful system, A1, has been chosen as a reference point, and its capacity assigned a value of 1. Based on the benchmark runs and extrapolations from them, the capacities of the remaining systems have been determined as multiples of the capacity of A1 and tabulated as in Table 5-a, where each column represents systems of comparable capacity. From the table, it follows that system A2 of manufacturer A is 1.8 times more powerful than system A1, and so on.

The rental prices of the various systems detailed above, including software charges, are shown in Table 5-b. Dividing the data of Table 5-a by those of Table 5-b provides an indication of the capacity per dollar outlay, which can be called cost-effectiveness, for each of the systems. These figures are shown in Table 5-c. Assume that the firm faces an anticipated growth in workload as shown in Figure 4. Year 0 on the horizontal axis is the current year, and year 1 is the year of installation. The planning horizon considered is six years. The vertical axis represents anticipated workload expressed as multiples of the workload capacity of the reference system, A1. The planned major expansion in

TABLE 5-a

Manufacturer	System 1	2	3	4
A	1.0	1.8	2.6	3.5
B	1.2	---	---	3.3
C	---	1.4	2.3	3.4

TABLE 5-b

Manufacturer	System 1	2	3	4
A	$60	$82	$100	$115
B	$77	---	----	$97
C	---	$82	$98	$116

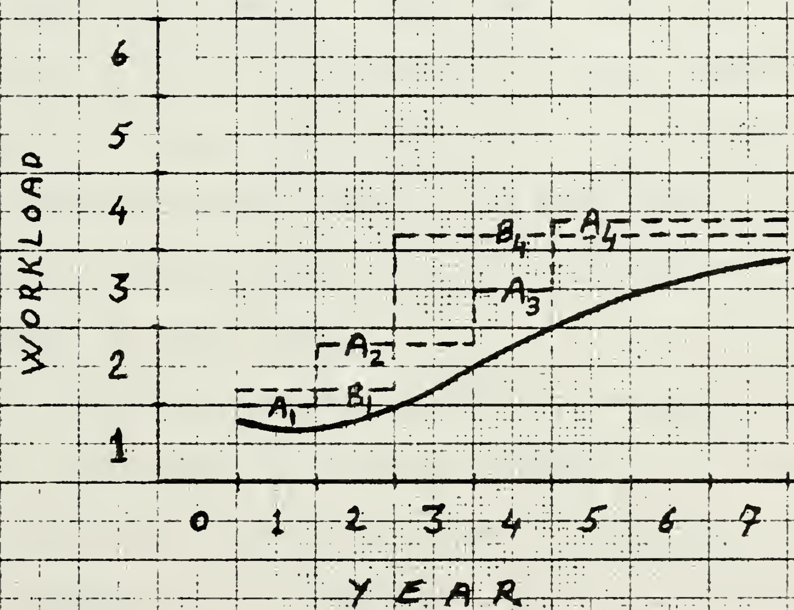
TABLE 5-c

Manufacturer	System 1	2	3	4
A	1.67	2.20	2.60	3.04
B	1.56	----	----	3.40
C	----	1.71	2.35	2.93

Ein-Dor 1977

FIGURE 4

THE GROWTH PATH FOR EACH OF THE SYSTEMS



Ein-Dor 1977

information processing activity causes a rapid climb in the curve in years one through five with a subsequent return to stability in year six.

At the end of six years, for this system, the workload will be about three times the capacity of the reference system. Under the conventional form of computer selection, a system should be chosen which either satisfies immediate requirements and is evaluated as having good growth potential or satisfies requirements for the next five or six years. The only systems which meet this requirement are A4, B4, and C4. This is illustrated on the right-hand margin of Figure 4. Since, from Table 5-b, system B4 is both the cheapest of the relevant systems and also has the highest figure of merit for cost effectiveness (see Table 5-c), it is the logical choice in this case.

The solution can be improved by using a modified approach to the upgrading of computer systems. It is generally conceded that there will be no more revolutionary change-overs between generations of computers, and evolutionary growth will become the order of the day. With respect to peripheral equipment, the evolutionary approach is already well established and most systems are progressively upgraded by the addition of new peripherals, or the exchange of lower capacity peripheral units for those of higher capacity. This is especially obvious with respect to discs and other mass storage units.

Given the compatibility provided between the processors and operating systems within the product line marketed by each manufacturer, it is now feasible to apply the evolutionary approach to cpu's as well as to peripherals. It is then possible to plan the replacement of a cpu after one or two years' service rather than after five or six years. If this approach is accepted, then one no longer selects a single computer system but rather one selects a series of computers within the compatible range offered by one manufacturer.

Assume that it is feasible to install a system for as little as one year, provided that it will be replaced by a compatible system from the same manufacturer's product line. Assume further that all installations are performed at the beginnings of years, and that system must be upgraded at the beginnings of years in which they would otherwise become saturated [Ein-Dor 1977]. Depending on these assumptions the growth path for each of the systems would be as in Table 6, and as exhibited in Figure 4.

The discounted present value of the rental systems, assuming a 20% cost of capital, is $3.15 million for manufacturer A, $3.50 million for B, and $3.78 million for C. Thus the dynamic evaluation presents A as the economically optimal solution rather than B in the conventional evaluation. The total undiscounted cash flow for this solution is $6.65 million compared to $6.98 for the static selection procedure.

TABLE 6

EXAMPLES OF THE GROWTH PATHS
FOR EACH OF THE SYSTEMS

Year	Workload (year end)	Manufacturer								
		A			B			C		
		System	Capacity	Rental	System	Capacity	Rental	System	Capacity	Rental
1	.85	A1	1.0	$ 720	B1	1.2	$ 924	C2	1.4	$ 984
2	1.10	A2	1.8	984	--	---	924	--	---	984
3	1.55	--	---	984	B4	3.3	1,164	C3	2.3	1,176
4	2.10	A3	2.6	1,200	--	---	1,164	--	---	1,176
5	2.65	A4	3.5	1,380	--	---	1,164	C4	3.4	1,392
6	2.95	--	---	1,380	--	---	1,164	--	---	1,392

The analysis above has been performed on the assumption that equipment is to be rented. The method is equally valid for purchase with purchase prices appearing in the analysis instead of rentals. Furthermore, if the dynamic analysis is performed for both lease and purchase, it can be of assistance in deciding on the method of acquisition. In using this approach for selecting the form of acquisition, one must take care to ensure that the values incorporated in the analysis are comparable. This requires that all relevant costs be factored in. For instance, where maintenance charges are included in rentals, they should either be added to purchase cost also or else rentals should be computed without maintenance. If purchased equipment is to be replaced, resale values should be determined and treated as negative costs. Any tax or excise differentials should also be taken into consideration.

Because of varying life expectancies of units within the projected system growth path, and because of variations in the ratio of purchase prices to rentals, it is almost certain that an economically optimal solution will indicate that some units should be purchased and others leased. This implies that the analysis should be performed stepwise. As each change is made to the system in the course of the analysis, the profitability of lease versus purchase should be evaluated; new units should thereafter be considered as acquired in the least cost mode.

This methodology is a basis for determining the relative cost-effectiveness of product lines in a given situation. This alone is not, of course, the only criterion for system selection. Other criteria such as software availability, manufacturer service, or reliability may well be at least as important. However, it is also wise to base one's decisions on as accurate a determination as possible of the cost factor.

The steps involved in the dynamic evaluation of cost-effectiveness overtime for a series of computer systems is as follows:

1. Determine relative capacities of relevant systems; possibly by benchmark runs.

2. Forecast the workload for the planning period in terms of a reference system.

3. Prepare a growth path for the systems proposed by each manufacturer with respect to the forecast workload and determine rental (or purchase) costs for each system.

4. Determine the discounted current value of each outlay stream, and so determine which manufacturer has the most cost-effective product line for the situation under study.

The application of this method may lead to considerably different results than the conventional methods of cost comparison. It determines a unique solution for a unique situation which is not generally transferable.

f. Present Value Analysis

A problem facing all buyers of computer equipment is deciding which system will cost the least and benefit his company the most. To determine if an investment in a proposed system will enhance the current value of a firm, present value analysis may be used. The present value of an amount to be received in the future is the equivalent value today of that future sum.

Because of the investment alternatives, future receipts should be discounted (their face value should be reduced) to an equivalent present value if they are to be compared with present receipts. For example, the present value of $126.25 to be received at the end of four years has a present value of only $100 if the investor has an opportunity to earn six percentage on invested funds. That is, $100 invested today at six percentage compounded annually will accumulate to $126.25 in four years.

Generalizing, the present value (PV) of a future sum can be determined using the equation

$$PV = \sum_{t=1}^{N} \frac{FV_t}{(1+K)^t}$$

where FV is the future sum, K is the opportunity cost (or rate of return), t is the year the future sum is received or paid and N is the number of receipts.

Since the decision to invest in a system should be considered independently of the method used to finance it,

the system should be evaluated as if it were being purchased for cash. The net present value (NPV) of the proposed system should be determined by discounting all the incremental, after-tax cash flows associated with it, using the firm's cost of capital as the discount rate. The cost of capital is the cost of new funds required to replenish the cash used for the purchase of the system. This process is expressed by the equation

$$NPV = \sum_{t+1}^{N} \frac{(R_t - C_t)\ (1 - T_c) + D_t T_c}{(1+K)^t} + \frac{S_N - (S_N - BV_N)\ T_c}{(1+K)^N} - I_o$$

where;

R_t: added gross revenue generated from the system,

C_t: the added cost of operating the system (operating costs do not include any financing cost such as interest or lease cost),

T_c: the firm's tax rate,

D_t: the depreciation,

t: the time period,

N: the number of years the asset will be economically useful,

S_N: the salvage value,

BV_N: the book value,

I_O: the initial purchase price,

K: the firm's cost of capital.

If the NPV is greater than zero, the system will add value to the firm because its rate of return is greater

than the cost of funds, and therefore it is a desirable project. When two systems are being considered, the one with the larger NPV should be chosen. If two alternative systems provide the same service or revenue (R_t), this item may be assigned a value of zero in the equation of NPV. In this case the NPV becomes negative and the system with the NPV closest to zero (the less costly alternative) should be selected.

The relevant benefits provided by the asset are the net cash inflows, which are discounted at the firm's cost of capital to obtain their present value. If the present value exceeds the cost, the asset is financially desirable because it adds value to the firm [Roenfelt and Fleck 1976, Szatrowski 1976].

VI. SYSTEM WORKLOAD DESCRIPTION

The most important part of any system specification is the part which describes the type and amount of workload to be run on the system. Performance is the degree to which a computing system meets the expectations of the person involved with it [Doherty 1970]. Performance is a reaction of a system to a specific workload. It is, therefore, essential that the right workload is used when evaluating the system and that the workload characterization is sufficiently representative to account for all significant factors. A good workload description should serve three important functions:

1. It should permit the vendors to determine what they need to propose for automatic data processing equipment and software (ADPE/S) to satisfy the workload requirements.
2. It should facilitate the verification of the proposed systems, both as to their capabilities to handle the workload, and as to the time required to complete the workload.
3. It should permit realistic costing of the bid systems.

The first two points, permitting determination of the proper system by the vendor and the verification of that system's capabilities, can be combined. If a good technique is achieved for the second purpose, that same method of workload description will also serve the first purpose.

User's applications, once translated into programs and commands, can be characterized by the type and the amount of resources the system will have to allocate to execute these programs and commands. The total of resource demands generated by the user community represents the system workload [Svobodova 1976, Joslin 1977]. Examples of parameters used to describe computer system workload are presented in Table 7.

In many computer installations, the instantaneous workload changes quite unpredictably. This is especially true for interactive systems. The speed of the user's response plays an important role in what load is generated at individual system entry points; this human factor only enhances the unpredictability of workload changes. It is this uncontrollable fluctuation of the system workload that makes the evaluation of system performance so difficult.

Generally, the workload of a computer system has certain statistical properties that do not change over reasonably long time periods. It is then possible to:

1. Characterize the workload by distributions of demands made on individual system resources.

2. Define a unit of work and express the workload as a number of such units.

Quantification of workload by work units is used when defining and comparing system processing capabilities. A unit of work is assumed to require a fixed but not necessarily explicitly known quantity of computation. Generally, it is very difficult to define a unit of work. Even when

TABLE 7

EXAMPLES OF WORKLOAD PARAMETERS

Workload Parameters	Description
Job CPU time	Total CPU time requested by a single job
Job I/O requests	Total number of I/O operations requested by a single job
CPU service time	CPU time requested to process a single CPU task
I/O service time	I/O time required to process a single I/O task
Interarrival time	Time between two successive requests for a system service
Priority	Priority assigned to a job by the user
Blocked time	Time a job is incapable of receiving CPU service
Memory requests	Amount of memory requested by a single job
Working set size	Number of pages of a single job that must be kept in the main memory
Locality of reference	Time for which all memory references made by a single job remain within a single page or a set of pages
User response time	Time needed by a user at an interactive terminal to generate a new request (think and type time)
User intensity	Processing time per request/user response time
Number of simultaneous users	Number of interactive users logged concurrently
Number in the system	Number of jobs or tasks being serviced or waiting in queues for system resources
Instruction mix	Relative frequencies of different types of instructions the system must execute

Svobodova 1976

programs are broken down to very elementary operations (e.g., instructions), the characteristics of such elementary operations differ.

A workload model serves as a workload of a real computer system during performance measurement experiments or as an input to a model of the evaluated system. The purpose of using workload models is to:

1. Provide representative workloads for comparative performance evaluation of different systems.
2. Provide a controllable environment for experimental performance optimization studies.
3. Reduce the quantity of data that have to be analyzed.
4. Present the system workload in a form required by a system model.

Alternate choices in system configuration and algorithms and the effect of different control parameters must be evaluated for the same workload. Generally, only one alternative can be examined at a time, thus requiring that the workload used as the input to the system during evaluation be reproducible.

The characterization of workload by demands made on system resources can also be used to define a unit of work. In fact, the workload parameters given in Table 7 are already defined with respect to a specific logical unit processed by a computer system. Such a logical unit is often adopted as a unit of work. To satisfy the requirement that a unit of

work represents a fixed quantity of computation, this logical unit is afixed with characteristics representing the mean of characteristics of all such units processed by the system.

The real system workload, that is, the workload generated by the user community in the normal production environment, is generally not reproducible in its exact composition. However, if the statistical properties of the system workload do not change with time, the workload is statistically reproducible. The real workload can be used to drive the system during evaluation experiments, but the measurement intervals must be sufficiently long, and it is necessary to collect and analyze large amounts of data to ensure that the statistics are correct. The minimum measurement interval may range from minutes or hours if the system workload does not change with the time of day, to weeks or months if the workload exhibits significant changes with such periodicity.

System workload may remain stationary for quite long periods of time, but in general, its characteristics change slowly as the user community changes because new applications are added and the old discontinued. In addition, the user community tends to adapt to system changes, and as the users change their habits, workload characteristics change. Thus in a long term, the real workload is not reproducible.

System workload is characterized by demands for system resources. Ideally, a workload model will have the same characteristics as the real workload. The model is accepted as being representative of the real workload if its application

results in the same steady state performance [Ferrari 1972, Svobodova 1976]. The lack of proper understanding of workload characteristics is a serious obstacle when the goal is to predict effects of system changes and design alternatives on performance. Extensive empirical studies of programs may reveal many interesting properties that should be considered during the initial system design. It is also important to study the habits of system users. Performance effects of certain system changes measured against a once representative workload model may be positive, yet in reality, the users may react to these changes in such a way that the overall effect will be negative. The true behavior of the eventual users may be quite different from the behavior assumed for the purpose of system selection or design [Warner 1972].

A system that is too carefully tuned to a specific projected workload might not meet the performance objectives if the real workload turns out to have different characteristics. It is thus necessary to have a means of examining performance in the light of different workloads. Flexibility and controllability of workload characteristics is an important property of a workload model.

A. INSTRUCTION MIX

An instruction mix represents the relative frequencies of different types of instructions a system must execute during a specified interval of time. The instruction mix specifies relative usage of different types of instructions in a particular

application. Since each instruction may require a different time to execute, performance (instruction execution rate) of an instruction set processor can be evaluated with respect to the requested instruction mix. Instruction mix is used in two main areas:

1. Selection of computer hardware,
2. Design of new processors.

In the first case, the typical instruction mix for the class of applications planned for the system must be defined such that it can be used across a wide range of different instruction sets. That is, a typical instruction mix specifies frequencies of different functions, rather than actual instructions that perform these functions. A typical mix might be in the proportion of five adds, two compares, one subtract, one multiply. This, then, might be described as a mix of instructions. By multiplying the frequency for which the instruction is typically used by the time a particular machine takes to perform the instruction and adding these together for the instruction mix, one can arrive at a figure which represents the time for the instruction mix on the particular machine. This figure can be calculated for various machines and thereby the machines compared for an instruction mix appropriate to a particular type of job. These figures can be arrived at from the theoretical timings for instructions of the machine or they can actually be measured by running a mix containing the appropriate number of instructions on the machine.

As instruction times are in milliseconds or microseconds, it is convenient to run the mix, maybe, ten thousand times in loop consecutively so that the start and finish can be measured in minutes on a stop watch. Some of these mixes have been commonly adopted as means of comparison. The most frequently used instruction mix is the Gibson mix, which can be classified as a "general purpose" mix. Figure 5 shows the calculations of the mix time and Figure 6 shows the approximate Gibson mix times in milliseconds for a number of machines.

Instruction mixes are also a means of comparing the speed of doing arithmetic in machines. Even so, this simple form of comparison may be prejudiced by dissimilarities between hardware which, perhaps, are advantageous to one machine and not the other. For example, word length varies between machines and to the user this is of some importance from the point of view of accuracy; i.e., in a process control application it may be perfectly adequate for the hardware to handle only four decimal digits, however for numerical analysis applications ten decimal digits may be required. Quite obviously a simple comparison of arithmetic operation is valuable only with other information about the specific applications [Graham and Yearsley 1973, Svobodova 1976, Gibson 1970, Joslin 1977].

Instruction mix depends on many factors that are difficult to account for, such as the number of operands per instruction or different addressing modes. Due to these factors, the number of instructions needed to run the same

FIGURE 5

CALCULATION OF MIX TIME

INSTRUCTION	MILLISECONDS	SECONDS
5 adds	10	50
2 compares	5	10
1 multiply	10	10
1 subtract	50	50
MIX TIME............................		0.120

FIGURE 6

APPROXIMATE PERFORMANCE FIGURES

PROCESSORS	GIBSON MIX MILLISECONDS
ATLAS	0.23
7094	0.28
1106	0.24
360/65	0.17
1108	0.11
360/75	0.11
1906A	0.10
6600	0.09
1110	0.025
370/195	0.02
7600	0.01
370/165	Approx. 0.07
370/155	Approx. 0.15
370/145	Approx. 0.30

Yearsley 1973

task on different machines may vary significantly. Also the instruction mix is dependent on the programming language in which the application is coded, the translator of this language, and finally the programmer [Lunde 1974, Svobodova 1976].

B. BENCHMARK PROGRAMS

A benchmark is defined as "a point of reference from which measurements can be made" [Sippl 1972]. A benchmark can be an instruction, a special program or a sequence of calls to selected software components. In most cases, however, the term benchmark is used to mean a job or a set of jobs that represent a typical workload of the evaluated system. Benchmarks play the role of a drive workload in the real system, both for the purpose of comparative evaluation of different systems and performance optimization. A good benchmark will exercise all system functions (job scheduling, file management, I/O support, language processor, etc.) in a manner in which these functions are used or are expected to be used in the actual production environment.

A benchmark representative of the current system workload can be assembled from already existing programs. Jobs to be included in the benchmark may be selected by random sampling of the job stream. This method does not require an explicit knowledge of characteristics of individual jobs, but it is then difficult to determine how many of these randomly-selected jobs must be included in the benchmarks [Shope 1970].

The real system workload generally consists of several classes of applications (scientific problems, payroll, file update, etc.). A benchmark or, as it is sometimes called, a benchmark mix, can be constructed as a properly weighted mix of jobs representative of each class. However, demand characteristics of jobs performing different functions may greatly overlap.

The most rigorous approach rests on partitioning jobs into classes according to their characteristics. The job with characteristics closest to the typical characteristics for its class is selected to represent the class in the benchmark. A selected job is assigned weights proportional to the percentage of workload that falls into that same category. Partitioning of jobs according to their true characteristics can be accomplished by cluster analysis. A clustering algorithm assigns jobs to a predetermined number of groups called clusters such that the differences between members of the same cluster are small compared to differences between numbers of different clusters.

A benchmark constructed from real jobs is apt to be system dependent. In general, such benchmark is not directly usable as a drive workload of a different system. A considerable conversion effort may be necessary to create a benchmark for several different systems [Joslin 1977, Svobodova 1976, Joslin 1965, Rosen 1976, Hunt, Diehr and Garnatz 1971].

There are several steps to obtaining the mix of representative programs to be used for benchmarking purposes. The following general guidelines should be kept in mind while searching out representative benchmark programs:

1. Where possible, benchmark programs should be written in a standard higher-level programming language; e.g., ANSI FORTRAN or ANSI COBOL.

2. The mix of benchmark problems should be small enough that it is capable of being processed during a single half-day benchmark demonstration.

3. The selected mix of benchmarks will demonstrate that the supplier's proposed system contains adequate memory and input/output devices, that the software proposed is operative and adequate, and that it has sufficient throughput speeds to the normal workload.

The benchmark programs are not to be selected to prove the worst case situation, but rather to demonstrate timing and capability for normal situation. If it is necessary to assure capability to handle worst case situations, benchmark programs selected for that purpose will be obtained; but they are not to be included in the mix; rather they will be treated separately as capability benchmarks.

The results of the benchmark will help in the evaluation effort by:

1. Proving that the vendor has a deliverable compiler and operating system.

2. Allowing the evaluator an opportunity to compare the relative speeds of the different compilers.

3. Determining the relative efficiency of the generated object code by comparing results of execution of the competing object programs.

4. Giving the evaluator sufficient information to allow determination of minimum internal memory requirements.

This last calculation is based on the size of the largest program which must be memory-contained at any one time, and is derived from:

1. The benchmark results, which allow determination of the ratio of object language instructions to Procedure Oriented Language (POL) statements, and the average size in terms of memory locations of the object instructions.

2. A user-generated estimate (in POL statements) of the size of the largest program to be core-contained. It is necessary to calculate the memory required for the program (MP) from the formula: MP = R x NS x IS, where:

R = ratio of object language to POL statements (from the benchmark),

NS = (estimated) number of POL statements for the largest program, and

IS = average instruction size determined from the benchmark by dividing the memory required by the number of object instructions.

The estimate of required internal memory is completed by

adding the program requirements, the memory requirements for the operating system resident, data and buffer storage, and communications-oriented subroutines.

3. Simple multiplication of the following factors:

$$\frac{\text{benchmark object code}}{\text{benchmark POL}} \times \text{estimated POL} = \text{estimated object code}$$

4. An estimate of the size of the average object instruction. This is obtained from the benchmark by dividing its memory need by the total number of object language statements.

5. Estimate of core for resident supervisor input, output buffers, and communications-oriented subroutines.

The summation of 4 and 5 will give an estimate of the required internal memory [Rubin 1971].

1. Derivation of Representative Programs

The following paragraphs describe a method for obtaining the representative programs.

a. Application

List each of the applications making up the total workload. This is illustrated in Table 8.

b. Programs and Tasks

For each computer program pertaining to the above applications, list the program and provide the information required in Table 8. For new programs or for acquisition of equipment that is for a new installation, it will be necessary to go through the normal design process with program flowcharts which lead to estimates of program run times, or to simulate the programs to obtain this information. Once the necessary estimates are obtained for

TABLE 8

APPLICATION, PROGRAM AND TASK INFORMATION

Application and Program Tasks	Individual Run Time (hours) (1)	Frequency Per Month (2)	Estimated Monthly Time (hrs) (3)= (1)x(2)	Subtotal (4)
Application A ADMINISTRATION				
A1: Time/Cost Studies	0.52	22.0		11.44
a. Sort			10.00	
b. Edit			1.44	
A2: Payroll Processing	0.25	4.3		1.08
a. Sort			1.00	
b. Validate			0.08	
A3: Transportation Usage				9.30
a. Compile - Cobol	0.50	1.0	0.50	
b. Extract	0.40	22.0	8.80	
....				....
A27: Miscellaneous				20.00
a. Sort			10.00	
b. Update			10.00	
Application B DISTRIBUTION CONTROL				
B1: Inventory Control	0.45	22.0		10.00
a. Sort			10.00	
B2: Allowance Generation				2.50
a. Compile - Cobol	3.00	1/6	0.50	
b. Extract	12.00	1/6	1.00	
c. Compute			1.00	
....				....
Application K MATRIX INVERSION				
....				....
Time Required on System			Total	880.00

Joslin 1977

Table 8; these new programs can be treated in the same way as any existing programs. Each program is broken down into its major functions or tasks, such as, sort, validate, update, extract, compute, card-to-tape conversion, tape-to-printer conversion, trajectory calculation, simulation, matrix manipulation, etc.

c. Task Summary

From each of the programs listed in Table 8 extract similar tasks and prepare a Task Summary Sheet for each task (see Table 9). Provide the information required in accordance with table headings which are explained below, which is the description of the columns in the Task Summary Sheet.

IDENTIFICATION. This column contains the code for each program in which the task is found. In the example shown in Table 8 the identification codes which would be given on a Sort Task Summary Sheet would be A1a, A2a,...A27a and B1a, etc.

I/O or FILE DESCRIPTION. This section is divided into four parts:

1. Media Code. Enter a mnemonic for the media that contains the I/O or file. Examples are:

MT Magnetic Tape

PT Paper Tape

PC Punched Cards

PR Printer

TABLE 9

TASK SUMMARY SHEET - DESIGN

The column headings are successively:

Identification

I/O or file description

- Media code
- Number of devices
- Category
- Block size

Monthly averages

- Frequency
- Volume
- Total time (hours)
- Peripheral equipment time
  - Magnetic tape
  - Card
  - Printer
  - Other

Internal storage requirements (in K's of storage)

Language

- Present
- Planned

---

2. Number of Devices. Number of devices that will be required for the use of this media which will have the same capability and category of use.

3. Category. Code designating the type or use of the I/O or file. The following codes shall be used:

0 Source or original input

1 Master file

2 Intermediate, working or scratch

3 Final output

4. Block Size. Product of the number of characters per record and records per block.

MONTHLY. This column is devided into four parts:

1. Frequency. Give the monthly run frequency of this program.

2. Volume. The number of blocks contained in the I/O or file. If this is a multi-tape file, follow the number of blocks by a slash and give the number of tapes. The volume to be recorded will be the average per unit, per month, for this task.

3. Total Time. Average total time to perform this task in the identified program. All times shall be given in hours and hundredths of hours.

4. Peripheral Equipment Time. Estimated average time required per task by each type of peripheral equipment. If similar units of differing capability are used, this timing information should be based on the highest capability available. Due to simultaneity and overlap, it is not expected that the total of the individual units will agree with the total time.

INTERNAL STORAGE. Estimated amount of internal storage required to process the task (not the full program, if the program is a multi-task program). If two or more processors are used for the task, enter the information accordingly.

LANGUAGE. List the language in which the task is programmed (first column) or is to be programmed (second column). If task is a self-contained library routine, the initials "L.R." should be entered.

TOTAL TASK TIME. At the end of the last Task Summary Sheet used for each type of task, there should be a line for total task time. The sum of these totals from all the Task Summary Sheets should equal total system time.

TYPICAL TASK. On the last entry of last Task Summary Sheet used on each type of task, there should be an entry for a nonexisting program. This entry should be weighted average (weighted by used time per month) for all previous entries for this type of task, and it should depict what a typical task of this type would look like.

d. Selection of Representative Tasks

From each of the sets of tasks, select tasks (preferably a single task program) which are representative of the set, or a substantial portion thereof, and identify these tasks with asterisks. The types and time of processing, amount of internal storage used, language used, and equipment configuration should all be taken into account when selecting the representative task; that is, it should be as nearly similar to the typical task as possible.

e. Extension Factors

A chart should now be prepared showing each of the representative tasks and the functions each presents. The monthly times required for each of these functions within a task should be listed alongside of the individual times of the benchmarks chosen to represent these task functions. The individual benchmark times for the functions should be divided into monthly times for these functions to obtain individual extension factors, which show how many times a month the representative benchmark would have to be run to make up the full monthly workload for the task. An example is shown in Table 10. If only sequential systems were to be considered, the individual functional extension factors would be sufficient, and each vendor could run the benchmark program, extend his system running time for each benchmark by the functional extension factors just derived, and thus be able to tell how long his system would take to complete the total workload.

In third-generation systems where many degrees of simultaneity must be considered, it is possible that while the system is handling one function, it could simultaneously be handling another, or even be multiply handling various tasks on programs. Thus, all the representative programs must be considered together to form the representative sample of the total workload. If the normal workload could be processed in variable ways, then a vendor should be permitted to handle the grouping or mix of representative benchmark

TABLE 10

REPRESENTATIVE PROGRAMS

Task Set	Workload Functions	Time (hours) Monthly Task	Time (hours) Representative Task (single run)	Extension Factor
Sort B-1a	Total thruput	145.00	0.45	322
	Mag. tape	125.00	0.25	500
	Card reader	115.00	0.03	3833
Edit E-4a	Total thruput	120.00	0.75	160
	Mag. tape	80.00	0.60	133
	Card reader	20.00	0.50	40
	Printer	100.00	0.25	400
Update D-5a	Total thruput	100.00	0.16	625
	Mag. tape	70.00	0.10	700
	Card reader	25.00	0.05	500
	Printer	50.00	0.10	500
Matrix Inversion K-6a	Total thruput	90.00	0.45	200
	Card reader	3.50	0.02	175
	Mag. drum	24.00	0.15	160
	Printer	1.50	0.01	150
FORTRAN Compile H-3a	Total thruput	85.00	0.17	500
	Mag. tape	78.00	0.15	520
	Card reader	6.00	0.02	300
	Printer	4.00	0.01	400
COBOL Compile G-2	Total thruput	40.00	0.12	333
	Mag. tape	38.00	0.11	345
	Card reader	4.00	0.04	100
	Printer	3.00	0.04	75
Tape to Print F-4	Total thruput	300.00	1.00	300
	Mag. tape	300.00	1.00	300
	Printer	300.00	1.00	300

Total Monthly Time........880.00

Joslin 1977

programs in whichever way his system can best handle them. If normally one type of workload would be handled before the other, then the mix should be structured in that way. However, just taking one each of the representative benchmark programs does not make a representative mix because the related extension factors also have to be considered. Table 11 shows a mix of representative benchmark programs and the mix extension factor.

f. Mix of Tasks

The extension factor for the mix is derived by examining the information contained in Table 10 and obtaining the lowest practical extension factor to reduce the number of problems to be run in the mix while retaining the required representative nature of the mix of problems, which in this case is 160. This extension factor is then divided into each of the sequential extension factors to obtain the quantity column. The previous column is then used to make the input/output total time when extended by the mix extension factor equal to the total projected input/output time. This mix of tasks can then be used as a proper demonstration of a supplier's multiprogramming or multiprocessing [Joslin 1977, Stimler 1974].

2. Expected Workload Levels

The workload to be processed by a system can be expected to increase over time. Therefore, the workload for a system can be envisioned as consisting of a series of various levels. These workload levels can be roughly

TABLE 11

A MIX OF REPRESENTATIVE
BENCHMARK PROGRAMS

PROGRAM	QUANTITY	PROVISIONS
B-1a	2	Normal
E-4a	1	Input from tape
D-5a	4	Twice normal
		Twice no output
K-6a	1	Normal
H-3a	3	Normal
G-2	2	Normal
F-4	2	Normal

Extension Factor for Mix: 160

approximated by the average monthly workload levels for each year of system life. The workload increases from one level to another as the system ages increase. Because of the uncertainties that are associated with projecting workload growth overtime, it is impossible to predict with complete accuracy just when the workload will reach a given level. Therefore, probabilities are used for this purpose as described below:

a. System Life

A chart showing system life should be prepared showing the number of years that the system is expected to be in existence (see Figure 7-a).

b. Projected Growth

A best-guess approximation of projected growth of the system should then be superimposed on the foregoing chart, the vertical axis depicting the workload in hours-per-month. Figure 7-b shows this.

c. Workload Levels

At the midpoint of the projected growth line for each year, construct a workload level line parallel to the horizontal axis (see Figure 7-c).

d. Level Probability

For each year of system life enter the probability of the average workload for that year being at or near each of these levels. The probabilities must be thought of as lumped at these levels in such a way that the total probability for a year adds up to 100%, because the number of levels

FIGURE 7 – a

SYSTEM LIFE

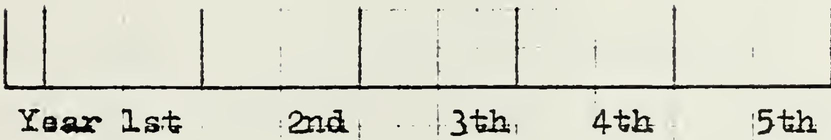


FIGURE 7 – b

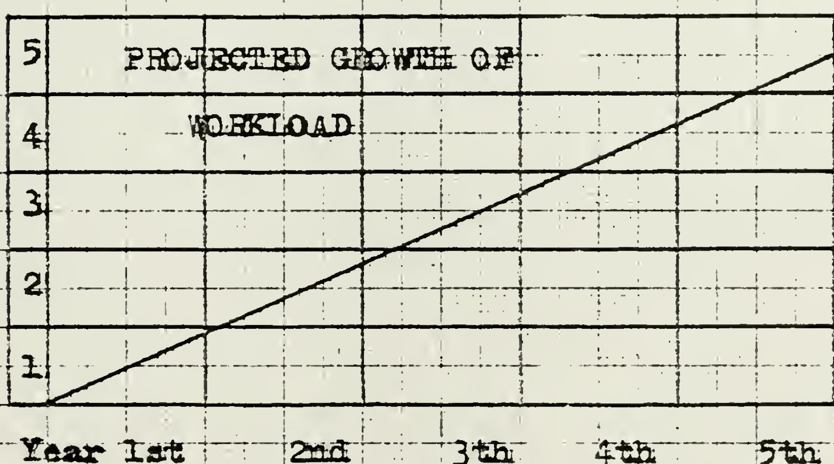
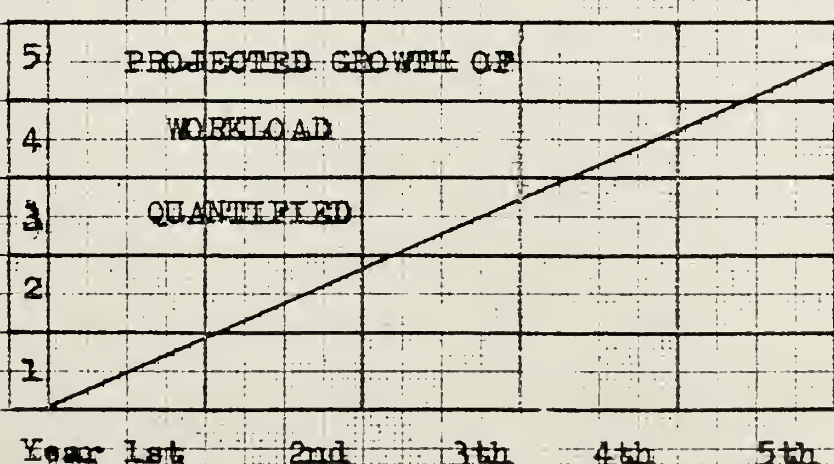


FIGURE 7 – c



used is finite [Joslin 1977, Svobodova 1976]. For example, referring to Figure 7-d, it can be seen that the workload there illustrated has a probability of 90% of being at level one for the first twelve months and a 5% chance of still being there for the second twelve months. Therefore, each vendor would be asked to determine the configuration necessary to process workload level one in the allotted time period, and to determine the monthly cost of that configuration.

FIGURE 7-d

EXAMPLES OF LEVEL
PROBABILITIES

Level:					
7	0	0	0	0	5%
6	0	0	0	5%	20%
5	0	0	5%	15%	70%
4	0	0	10%	75%	5%
3	0	10%	80%	5%	0
2	10%	85%	5%	0	0
1	90%	5%	0	0	0
Year:	1st	2nd	3rd	4th	5th

VII. METHODS OF PROCURING COMPUTER SYSTEMS

The three most commonly used computer procurement plans offered by the vendors are lease, purchase, and lease with option to purchase. The decision on the selection of computer hardware (and software) and procurement methodology is a management responsibility, and should be based upon a feasibility study and subsequent evaluation process. In connection with hardware selection, the manager always makes a second decision; that is the decision as to whether to purchase the computer or to rent it. The feasibility study should include recommendations on the purchase-rent decisions and the facts upon which these recommendations were based.

A. COMMON PROCUREMENT PLANS

In this section the common methods of procuring computer systems will be introduced.

1. Leasing

Leasing, in the context of computer use, usually means an operating lease, with ownership of the computer system retained by the vendor. The user pays a predetermined monthly price for the use of a certain length of time on the computer system. The lease price includes rental of the equipment, a fee for the maintenance and service of the equipment, and a payment to compensate the vendor for the risk of ownership. The length of the lease is very important in its effect

on the rate. Since the leases which are proposed by the manufacturers are usually short-term ones, the following discussion will concern itself with these short-term leases.

Under a normal short-term operating lease, the user enjoys the following advantages:

1. Frees working capital for more productive use (since money is not tied up in low-yielding fixed assets).

2. May cost less than other methods of acquiring equipment.

3. May increase the firm's ability to acquire funds.

4. Establishes only a restricted (not a general) obligation against the company which may be satisfied by payment of one year's rent in bankruptcy or three years' rent in reorganization.

5. Does not appear as a liability on the leasee's balance sheet.

6. Leaves normal lines of bank credit undisturbed.

7. Permits 100% financing (as against 75% or 80% through other methods).

8. Creates an allowable cost (or acceptable cost according to the government regulations including interest cost) under government contracts.

9. Permits hedging of business risk (primarily the risk of obsolescence).

10. Minimizes danger of being oversold.

11. Assures more adequate servicing (since maintenance is the responsibility of the lessor and usually included in the lease contract).

12. Offers the convenience of making only one payment (rather than separate payments for debt service, maintenance cost, insurance, property taxes, etc.).

13. May be tailored to the leasee's computer system needs more easily than ordinary financing.

14. Avoids the necessity of selling equipment no longer wanted.

15. Permits middle-management executives to acquire new equipment without going through formal appropriation request procedures.

16. Provides cost-cutting equipment to be installed immediately.

17. Acts as a hedge against inflation.

18. Provides long-term financing without diluting ownership or control.

From the point of view of the leasee, equipment leasing has the following disadvantages:

1. Equipment leasing charges a higher interest rate (than the leasee's regular interest rate).

2. May provide less attractive tax deductions (than interest plus accelerated depreciation).

3. Gives any residual value of the equipment to the lessor.

4. Establishes a fixed obligation against the company.

5. Does not provide whatever prestige that goes along with ownership.

6. Raises the fear of dispossession if payments are not made during hard times.

The importance of the listed advantages depends on the individual company and the environment in which the computer is to be used. System obsolescence within the organization that is using the computer has proved much more significant than technological obsolescence. The knowledgeable prospective user, in forecasting the life expectancy of the proposed system, should carefully study and evaluate his data-processing needs. However, since every user does not exercise the same degree of foresight in planning, the vendor must set his lease charges so that they allow for the average system life expectancy. This is a compromise measure since the vendor must deal both with those users who plan and those who do not. One can therefore see that if the user has done a good job of planning his systems, he is in a better position to assume any risk of system obsolescence than the manufacturer is. Also, it can be costly for the user to lean on obsolescence as a crutch or to allow it to influence the lease/purchase decision.

Another important disadvantage of leasing is that on a leased computer system, extra usage is more expensive than it is on an owned system. And if the user makes any serious attempt to make the computer pay for itself, he may have to utilize these extra shift hours [Joslin 1977, Vancil 1962, Gustafson 1973].

The concept of the third-party operating lease on computer equipment originated in the United States. Indeed, even in Europe, the service has been provided almost

This page intentionally blank

exclusively by US-based companies. Its origin can be traced as far back as 1961 when D. P. Boothe Inc. wrote an operating lease on an IBM 7094 for Ling-Temco-Vought. The principle attraction to the user was a saving in additional use charges, then 40% of the primary shift rental.

Second-generation computers were not really amenable to leasing because any significant increase in power or capacity required a change of hardware, so that even a medium-term commitment would not have been tolerable. This constraint disappeared with third-generation systems which permit considerable growth through addition rather than change. At the same time, the total cost of running a computer was steadily mounting; because of the increased power and sophistication of the equipment, correspondingly more had to be spent on software and other supporting functions. The overall cost meant that the user had to think in terms of a longer life span for each system he installed, in the range of say three to five years [Gustafson 1973, Szatrowski 1976].

The hardware itself is now sufficiently reliable, flexible and modular for a functional life span of at least ten years to be foreseen, against four years for rental to become equivalent to the purchase price. Being prepared to wait longer than this to recover their costs, the leasing companies could provide the equipment at less than the manufacturer's rental.

The subject of computer leasing revolves almost entirely around IBM computers. Because of IBM's large market

share, early in the game the computer lessors elected to purchase IBM equipment since they felt it had the best chance of being placed elsewhere. A company with a small market share would create unacceptable risks to the lessor. The initial success of computer lessors so far as their ability to raise large quantities of capital and to convince many users of the viability of the concept is well known. But changes in IBM policy curtailed their growth.

For many years IBM was a one-price shop. That is to say, it made no difference whether you used one computer or ten, your unit price was the same. It made no difference whether you could use the equipment for five years, ten years, or one year. Your price was the same. It made no difference that technical requirements that you could project did not indicate or consider significant growth. Accordingly, many users were subsidizing the requirements of the more sophisticated user. Computer leasing then became a viable alternative because the computer lessors were able to offer leasing programs that more closely matched a customer's requirements.

The discount offered by the computer lessors is the obvious advantage to the user as compared with leasing from the manufacturer. Services provided by computer lessors vary substantially from company to company. Some provide services on the theory that they will enhance the ability to move equipment around. Others got into other services for diversification reasons with the expectation that they would be getting into other profitable businesses. For example, at

one time Randolph Computer Corporation owned a number of data processing centers in the Pacific Northwest and in the Mid-west. These provided a whole range of services normally found in such centers but really had no direct relationship to the computer leasing activity. It gave a reservoir of skills in programming and systems engineering that could be used from time to time in the leasing activity. However, only very rarely were these skills found to be required.

As time propressed, computer lessors have recognized that they cannot look at themselves exclusively as offering a financial service, that is to say, renting a computer at a price that is less than the user would pay IBM. The business has become increasingly technical in nature. It is obviously beneficial if it can be demonstrated to a user that there are more effective equipment configurations to meet his requirements. This is advice that he will not always get from the manufacturer since it sometimes implies less equipment [Yearsley 1973, Gustafson 1973, Randolph 1974].

It can be told to the customer how efficiently his equipment is being used by the use of a hardware monitor. Then the customers must frequently be assisted in upgrading from one model to another. For example, upgrading from a 360 model 20 to a model 30 has many software ramifications and requires a considerable amount of handholding.

There are some risks to a company doing business with a computer lessor. As with maintenance, in the early days of computer leasing there was some fear on the part of the user

that by cutting the umbilical cord to IBM, he might be cutting himself off from valuable services. It is alleged that there have been instances where the salesman on an account may have implied that this was so. This was basically contrary to the ground rules under which IBM is supposed to operate. Fortunately, these instances have not been frequent. The more important risks lie in the area of whom one chooses to do business with. Some of the computer lessors have gotten themselves into financial difficulty as a result of unwise diversification efforts. Others have withdrawn from the computer leasing field because they find that their diversification efforts have been so successful that they elected to concentrate on them rather than on computer leasing. Customers doing business with such companies obviously do so at their peril.

It is important that a computer lessor be chosen for its financial stability and its demonstrated ability to stay in the business for the long haul. Flexibility that customers require can be met only by someone who is wholly committed to this business [Gustafson 1973, Sabol 1972].

It is also important that the equipment be maintained in the best condition. Routine maintenance is, of course, an important part of accomplishing this. In addition, when equipment is moved from one customer to another, it frequently goes through a refurbishing center where catch-up maintenance is performed. The pressures of day-to-day work at an installation often will not permit all the maintenance routines to

have been effectively completed. This can be accomplished at a refurbishing center. It is also important that installation be performed smoothly and with a minimum of disruption. Proper preinstallation planning is an essential ingredient. One of the measures of the effectiveness of any computer leasing organization is how well it contends with emergency situations as they occur [Randolph 1974, Oliver 1973].

a. Lessee Motivation

However mixed the options of the manufacturers, however varied the fortunes of the leasing companies, the computer user who employed a leasing facility has generally found it highly rewarding. Without any significant change in his relationship with the supplier, the user has been able to obtain exactly the same equipment for between 10 and 25% less than the normal rental charge [Graham 1973].

Substantial cost savings have, therefore, been the principle benefit to the user. Also, leasing offers another option in the choice of acquisition method. Traditionally the manufacturer offered two alternatives: rental, usually for a minimum of twelve months only, or outright purchase. Leasing provides a further choice: a lower rental charge for a longer period of commitment. The previous rental user and the previous purchase user both found the leasing proposition attractive. This is because the conventional alternatives are best only in extreme situations which are not the normal user requirements: short-term

rental is suited to the user who wishes to make frequent major changes to his equipment, and purchase to the very stable environment where a commitment of six years or more is acceptable and where, also, capital and credit are readily available and not better employed for other purposes.

In practice, even for the rental user, major equipment changes cannot be economically made in less than two to three years so that the flexibility provided with a twelve-month agreement is more illusory than real; it is in fact a relic of the punched card era when change was less pervasive in its effect on a company's overall activities.

The purchase user already realised that the manufacturer's rental terms offered flexibility he did not need at a price he did not want to pay. However, the purchase alternative contained two deterrents: first, commitment to hardware over a period long into the future (at least six years) during which unforeseen requirements could arise for computer processing power, and during which technological advance could make the equipment obsolete; and second, tying up large amounts of capital or credit which could normally be employed better elsewhere in the company, in its own line of business.

Leasing, therefore, provided a very acceptable compromise between the extremes of rental and purchase, and the commitment of two to five years, tailored to the user's plans, was less of a hardship than a correlation with real requirements. Furthermore, different components of the

system could be rented or purchased if these methods of acquisition were selectively found most suitable. For example, if a faster printer is planned one year after the main installation, this could be rented direct from the manufacturer, while the rest of the system is leased.

In addition to the saving against the manufacturer's normal rental charge, leasing contains a further tangible advantage. The leasing company's charge normally covers use of the equipment 24 hours a day, whereas the manufacturer's standard rental is for a specified number of hours per month, roughly equivalent to a single shift five days a week. An additional charge is made for use beyond this period. It is true that the leasing customer will probably incur an additional charge for maintenance, but this again is less than the rental alternative [Graham 1973, Coutinho 1977, Tatham 1969].

There are a variety of ways in which the leasing customer can capitalize on the benefits available. Most obviously, he can reduce the cost of his computer installation. Alternatively, he can have more equipment or more people for the same expenditure as previously budgeted under the manufacturer's rental plan. In addition, there are more far-reaching opportunities for the user, the benefits of which could far outweigh the direct savings. Through leasing, it may well be possible to install a system of greater power than originally envisaged, and then keep it for longer. A leased 360/40, for example, may cost as little as a 360/30 in about

two years. The user who needs the power of a Model 30 now and a Model 40 in, say, two years' time could afford to install the Model 40 at the outset and then hold it for five years [Yearsley 1973].

The real advantage from doing this is not only in acquiring more power for the same money, but rather in conferring stability on the data processing departments. By avoiding frequent changes in hardware the user avoids the concomitant expense of software and procedural changes. Where there are frequent major changes in equipment, the energies and costs of the data processing department are expended on technical transition from one language to another or from one operating system to another. This does not make profit for the company. However, by first establishing a stable technical environment for, say, five years, the data processing staff can then concentrate on its main purpose of increasing the computer's functional contribution to the company's business. In this way, the leasing facility provides the opportunity not only for better value from expenditure on the hardware itself, but for better value from the whole computer investment.

Major companies tend to predominate among the customers of the leasing companies for a number of reasons: they have more to gain in absolute terms, they were quick to perceive the advantages, and they were attracted to the leasing companies because of their credit standing and because they tended to have medium to large computer systems which most lessees favored [Randolph 1976, Graham 1973].

b. Lessor Motivation

Like the lessee, the lessor is in the business for the money he can make out of it. Already a number of entrepreneurial fortunes have been made (and some lost) in the USA from this business and some companies of substance have emerged, already diversified into other fields.

The lessor's starting point is his willingness to take an eight- to ten-year view of the computer as a revenue-earning investment. He supports this position in a number of ways: first, the computer is an electronic device with little to wear out; second, third-generation computers are sufficiently reliable and modular to have a long working life; third, they have proven to be compatible with the next generation of hardware; fourth, the pace of technical change in the computer industry is slowing down.

The lessor's view of the machine is thus quite different from the user's. It is also quite different from that of the manufacturer's. In developing and building a computer, or family of computers, the manufacturer has invested huge sums of money. Even before the first computer of a new 'generation' reaches its first customer the manufacturer has spent millions of dollars on research and development, and on plant and equipment. He has to recoup this within as short a period as the market place will allow, in a market place which is rental-oriented. In practice the selling price of a computer is normally recovered in approximately four years of rental [Bucci 1973, Borovits 1975].

So the manufacturer's rental-to-purchase ratio is four years, the user thinks in terms of three to five years, and the lessor believes it will earn reasonable revenues for eight to ten years. Here then lay the opportunity for providing an attractive service which could itself become a substantial business. Leasing gained rapid acceptance and generated large and fast-growing profits for the lessor. These were to some extent dependent upon the rate of depreciation, and true profits would be obtained only when the lessor had fully recovered all expenses at some time in the future, based on the ability to remarket equipment when the first user had finished with it. This remarketing capability certainly did not exist, nor was it needed, when the initial leases were being written. The lessor would certainly need this capability and/or other business activities to offset the risk inherent in the leasing operations themselves.

The leasing companies did not have to wait long to satisfy these requirements. The size of profit they were generating and their rate of growth rapidly captured the imagination of the investing public in the USA, providing a high multiple for the company's stocks. This in turn gave the leasing companies the opportunity for acquisition and diversification [Yearsley 1973, Oliver 1973].

c. Lease Contract

The lease contract contains characteristics of both the purchase and rental contracts. In the computer industry, lease contracts are available through "third-parties", or directly from the vendors. The third-party company will purchase the equipment from the manufacturer and lease it to the user. The terms can be flexible and negotiable, depending on the risk to the lessor; thus the longer the duration of the lease, the more favorable the terms and conditions possible to the user. The lessor must rely on the cash inflow (depreciation tax deduction plus cash payments) and the residual value of the equipment to cover his costs. If the term of the agreement is of relatively short duration, the lessor must look forward to the problem of finding a second user.

Lease contracts fall into two general categories:

1. Full payout or financial leases.
2. Non-full payout or operating leases.

In the full payout or financial lease, the user (or lessee) essentially has the rights of purchase and assumes the risks normally assumed by the purchaser. The legal title, however, is retained by the lessor. The lessee's payments are designed to recover for the lessor:

1. The total cost of the equipment.
2. The cost of money required to purchase the equipment by the lessor.
3. A contract fee, normally about 0.5% or more.

At termination, the lessor still owns the equipment, although the lessee will normally have the option to purchase. The full payout lease is normally used to obtain financial benefit for the lessee; for example, lower payments over the useful life of the computer as compared to a rental [Szatrowski 1976, Gustafson 1973, Bucci 1973].

The non-full payout or the operating lease has many characteristics of a rental contract. The essential difference is the length of commitment. The term of this contract generally starts with a minimum commitment of two years, and can go as high as ten. Monthly payments average 10% to 30% less than the manufacturer's rental price. Generally speaking, a lease contract (either financial or operating) is the most flexible of all contracts abailable to a user of computer equipment. The user can negotiate with the lessor for terms most beneficial to both parties. These negotiations are somewhat unusual since both parties, by and large, are aware of each other's financial needs and requirements. Some items that affect the negotiations are:

1. Maintenance
2. Depreciation
3. Investment tax credit
4. Property taxes and insurance.

One of the two parties must pay for maintenance, and the cost is the same for either party. There may be local advantages for one party or the other to assume the maintenance obligation. For example, the user may already

have a maintenance contract with the manufacturer for other computer equipment and could perhaps extend it to include the leased equipment. Alternatively, the lessor may have a national contract with the maintenance organization.

The investment tax credit is a direct tax benefit for one of the parties. In certain cases, it could benefit one corporation more than another. For example, if one of the companies may be operating in a loss period, it may not need the investment tax credit since its tax would not be as large as in other periods. Another case might occur when a company makes massive investments, say an airline in the years it purchases new planes; such investments may exhaust the potential investment tax credits. In such cases, by relinquishing the investment tax credit, the user may be able to negotiate a lower lease price.

There are some additional tax considerations to be taken into account in a leasing arrangement. For a transaction to be acceptable as a true lease, i.e., not as an installment purchase contract, the lessor is required to assume a significant risk both during the lease term and in the period after its expiration. According to IRS regulations the ideal lease arrangement would have characteristics among which are these:

1. Lease payments would be approximately the same throughout the basic lease term.

2. Purchase options are not at fixed amounts but are based on fair values at the end of the lease term.

3. The estimated fair market value of an asset at the end of the lease term is at least 10% of the asset's original cost.

4. The lease term is less than 80% of the asset's useful life.

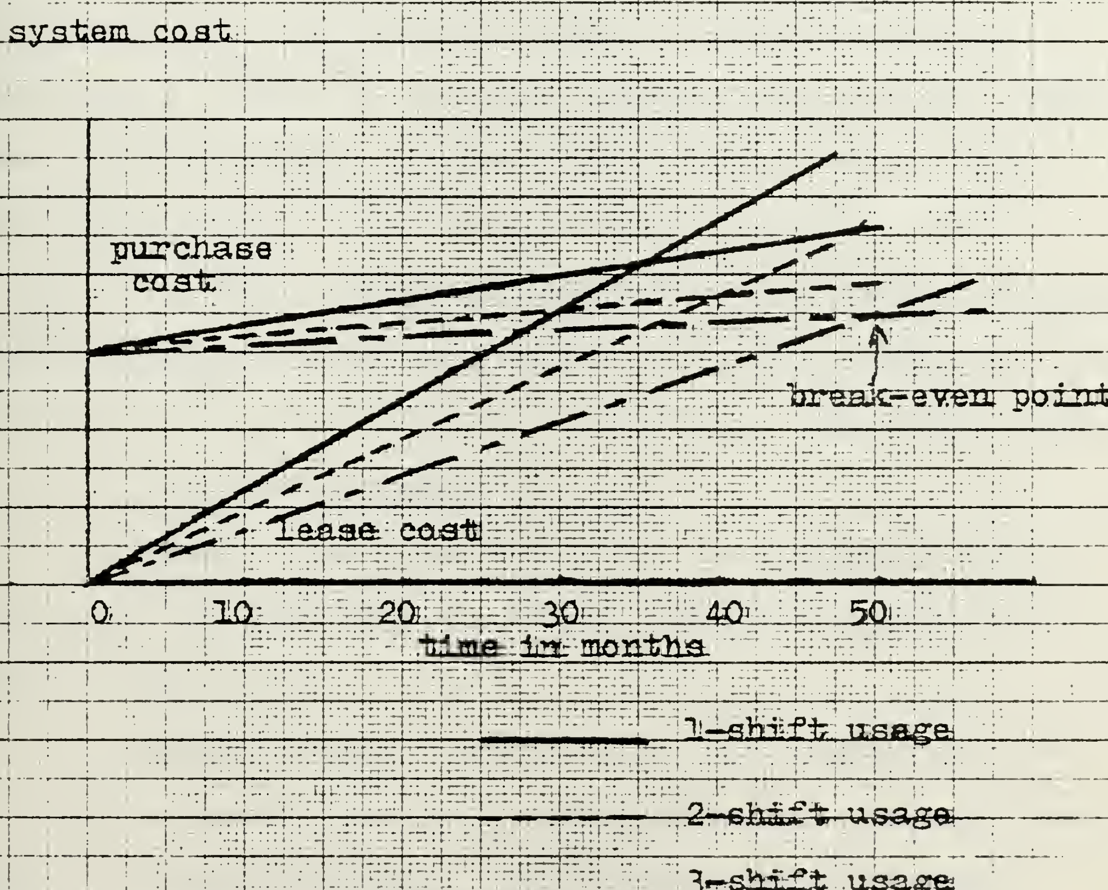
Also important in the financial analysis of the lease contract is the unlimited availability of the equipment for the lessee. There are also no overtime use payments associated with a lease contract [Szatrowski 1976, Gustafson 1973, Sabol 1972, Randolph 1974].

2. Purchasing

When the user purchases the computer system, he acquires ownership of it and can use it one shift or around the clock, seven days a week, with minimal increase in hardware expenses. Maintenance and service of the computer are contracted for separately with the vendor. With respect to taxes, the user may depreciate the purchased computer system as he would any other item of capital equipment. Any operation of the system beyond the break-even point constitutes pure profit to the user-owner, for he avoids those lease payments which he would be making had he decided on an operating lease [Randolph 1974, Joslin 1977, Graham 1973].

The break-even point can be calculated as follows: The number of months to break even equals the purchase price divided by the difference between the monthly lease cost and the monthly maintenance cost; (see Figure 8)

FIGURE 8



Joslin 1977

$$BE_m = \frac{P_p}{LC_m - M_m}$$

BE_m = the number of months to break even

P_p = the purchase price

LC_m = the monthly lease cost

M_m = the monthly maintenance cost

The user stands to enjoy certain tax benefits, but he assumes the normal risks associated with ownership: If the system fails, the responsibility is his and not the manufacturer's. When he considers purchasing, the user ought to take the time value of money into account: Money spent today is more costly than the same amount of money spent some years from now. Given an interest rate of 10% per annum, one million dollars used to purchase a computer today has the same value as 1.6 million dollars spent five years from now.

a. Purchase Contract

Under a purchase contract, the purchaser bears all the risks of ownership including insurance, taxes, and equipment obsolescence.

By and large, the purchaser will obtain the same services and support from the vendor that are available under a lease or rental agreement. There are, however, three important factors affecting this financial decision:

1. Full payment must be made to the vendor upon delivery of the equipment.

2. A separate maintenance service contract must be negotiated since service of the equipment is not considered part of the purchase price.

3. Insurance premiums and appropriate taxes must be paid on the asset.

Assigned values of depreciation can substantially affect the cash flow analysis for a purchased system. The buyer of any expensive capital equipment should be acquainted with the optimum depreciation schedules allowed by law. In addition, the future projected tax position in the corporation should be considered in order to be able to calculate its after-tax cash flow.

The assignment of a residual (or market) value to the equipment at some future date is probably the most difficult estimate to make in the financial analysis. If the residual value is too optimistic, losses are experienced at resale or trade-in time. On the other hand, assigning a zero dollar value as residual may be entirely unrealistic. Under such circumstances, it may be advisable to assign both the most pessimistic and the most optimistic value for residual, with analysis under both conditions. Statistically it may be possible to determine the most probable outcome under these circumstances [Szatrowski 1976, Joslin 1977, Randolph 1974].

b. Rental Contract

Under the rental agreement, the user is liable for a prepaid fixed minimum payment. The agreement can be

terminated by a minimum of 90 days prior written notice. Under this agreement, the risk of ownership remains with the vendor. The user has no obligation for such expenses as insurance and maintenance; however, he is responsible for paying taxes that might be levied on the rental contract by the state or local government.

Extra shift use, over and above the standard monthly base hours, represents an additional cost to the user. Investment tax credit is also a consideration under a rental contract and can be passed to the user. Rental contracts find a high level of usage in the computer industry due to a number of factors:

1. Low risk
2. Financial leverage
3. Equipment obsolescence
4. Flexibility

Flexibility is probably the best argument for a rental contract. When the user has a continually varying mix of jobs that require different configurations of equipment, it is to his advantage to be able to move equipment rapidly in or out of the installation without penalty charges.

Straight purchase has two serious drawbacks:

1. It requires a relatively large sum of money all at one period of time.

2. It does not permit the activity to adequately test the equipment and their system before they have committed themselves to buying something which may turn out to be too large or too small.

Straight purchase plan should be used only when the system life, including reuse, is longer than five years and the equipment has had ample time to demonstrate its ability to handle a proven application and purchase money is available [Szatrowski 1976, Coutinho 1977, Gustafson 1973].

In 1977, a revolution was taking place in the prices of medium and large-scale computer systems. Prices for hardware fell dramatically. Table 12 gives an idea about these prices.

3. Leasing with Purchase Option

The main advantage of the lease with purchase option is that there is a trial period in which the manufacturer's system is tested in the user's applications. If the system cannot satisfy the requirements of the applications, it can be replaced at relatively little expense to the user. If, however, the system fully satisfies the requirements of the applications, the user can carry out the purchase by exercising the option and buying the system. In this case, little money will have been wasted since the larger part of the total lease payments can be applied to the purchase price.

Sometimes the purchase option has to be negotiated separately with the manufacturer, and sometimes it is a standard part of the lease contract. The intent of this option is to give a user the right to purchase the system within some specified period of time, normally one or two years. If the right is exercised, some stated percentage of the money paid to the vendor is applied toward the purchase

TABLE 12-a

EXAMPLES OF COMPUTER SYSTEMS' PRICES

Characteristics \ Modal	IBM 3031	Itel AS/5 7031	Burroughs B6817	CDC Cyber 173	NCR V8590	Univac[7]		HIS[11]	
						90/80-3	1100/80	68/80	68/DPS Level 1
Purchase Price CPU and 2M Bytes (No. of Channels)	1,000,000 (6)	1,100,000 (7)[2]	975,000 (40)	1,196,920 (12) 98,000 (60-bit) words	806,640 (6)	1,035,540 (6)[8]	1,447,670 (6)	3,337,200 (24)	1,312,913 (24)
Price/Mo With Maintenance (Lease Term In Years)	25,000 (4)	Not Available	23,515 (5)	25,687 (5)	20,718 (5)	22,813 (5)	28,060 (5)	65,385 (5)	33,045 (5)
Maintenance/Mo[1]	3,070	Not Available	4,551	3,621	3,017	2,733	2,420	7,058	4,253
Memory Size In Megabytes (Minimum-Maximum)	2-6	1-8	1.5-9	65,000-262,000 60-bit words	2-6	2-4	2-4	1-8	2-16
Memory Cycle Time (Nsec)	345-805[13]	Not Available	325[3]	400[4]	475[6]	457-588	600-100[10]	See Footnote 12	See Footnote 12
Machine Cycle Time (Nsec)	115	100	Not Available	50	56	98	50	Not Available	Not Available
Channels (Minimum-Maximum)	6	7	40	12-24	4-24	4-8	6	56-224	56-224
Price/Channel					14,440	15,840[9]		Not Available	Not Available
Memory Buffer	32K	32K	No	See Footnote 5	No	No	16K	2K	2K

CW Chart

1. IBM maintenance is minimum monthly charge, 10 hour/day, 5 day/week. Burroughs maintenance is 24 hour/day, 7 day/week.
2. Seven standard channels, six operational.
3. Read access is 325 nsec.
4. Effectively 50 nsec.
5. Scheduling arrangement for peripheral processor units and CPU to access main memory. Buffer could be one to 12 CDC's 60-bit words. System quoted has 10 peripheral processor units.
6. Also features four-way interleaving.
7. Univac prices include software.
8. Five block, one byte multiplexers.
9. For each block multiplexer.
10. Speed is 650 nsec for 16 bytes in "rear memory" and 100 nsec from "forward memory" to CPU.
11. HIS expects to offer other systems in this range based on Level 66 with Gcos operating system. The 68/DPS units have Multics operating system. The 68/80 is what HIS would have bid prior to October 1977; the 68/DPS is the firm's current offering.
12. Memory access time is 750 nsec.
13. Depends on type of work being done.

Computerworld 1978

TABLE 12-b

EXAMPLES OF COMPUTER SYSTEMS' PRICES

Model / Characteristics	IBM 3032	Amdahl 470V/5	Itel AS/6 7032	Burroughs B7811	CDC Cyber 175	Univac[6] 1100/81	Univac[6] 1100/82	HIS[8] 68/80 (2 CPUs)	HIS[8] 68/DPS Level 2
Purchase Price CPU and 4M Bytes (No. of Channels)	2,142,000 (6)	2,650,000 (8)	1,960,000	2,811,540 (28) For 4.5M Bytes	3,353,623 (12) 131,000 60-Bit Words	1,821,690 (6)	2,547,410 (6)	4,010,130 (24)	2,099,467 (24)
Price/Mo With Maintenance (Lease Term In Years)	51,360 (4)	72,875 (4)	Not Available	56,555 (5)	69,425 (5)	35,135 (5)	59,585 (5)	79,287 (5)	50,889 (5)
Maintenance/Mo[1]	6,820	9,150	Not Available	7,100	7,568	2,875	4,530	8,662	6,467
Memory Size In Megabytes (Minimum-Maximum)	2-6	2-6	2-16	3-6	131,000-262,000 60-Bit Words	2-16	4-16	1-8	2-16
Memory Cycle Time (Nsec)	320	300	Not Available	See Footnote 2	400[4]	600-100[7]	600-100[7]	See Footnote 9	See Footnote 9
Machine Cycle Time (Nsec)	80	32.5	72	Not Available	25	50	50	Not Available	Not Available
Channels (Minimum-Maximum)	6-12	8	6-16	28	12-24	6-52	6-52	56-224	56-224
Price/Channel	360,000 (6 channels)		50,000			54,075 (4 Channels or 1 Block Multiplexer)	54,075 (4 Channels or 1 Block Multiplexer)	Not Available	Not Available
Memory Buffer	32K	16K	64K	12K[3]	See Footnote 5	32K	64K	2 Buffers Of 2K Each	2 Buffers Of 2K Each

CW Chart

1. IBM maintenance is minimum monthly charge, 10 hour/day, 5 day/week. Amdahl and Burroughs rates cover 24 hour/day, 7 day/week.
2. Access time is 50 nsec/byte.
3. Consists of 12K program buffer plus data buffer and storage as well as central data register.
4. Effectively 50 nsec.
5. Scheduling arrangement for peripheral processor units and CPU to access main memory. Buffer could be one to 12 of CDC's 60-bit words.
6. Univac prices include software.
7. Speed is 600 nsec for 16 bytes in "rear memory" and 100 nsec from "forward memory" to CPU.
8. HIS expects to offer other systems in this range based on Level 66 with Gcos; the 68/DPS units have Multics operating system. The 68/80 is what HIS would have bid prior to October 1977; the 68/DPS is the firm's current offering.
9. Memory access time is 750 nsec.

Computerworld 1978

TABLE 12-c

EXAMPLES OF COMPUTER SYSTEMS' PRICES

Characteristics / Model	IBM 3033	Amdahl 470V/6-II	Amdahl 470V/7	Burroughs B7321	CDC Cyber 175	Univac[6] 1100/83	Univac[6] 1100/84[8]	HIS[9] 68/80 (4 CPUs)	HIS[9] 68/DPS Level 5
Purchase Price CPU and 6M Bytes (No. of Channels)	3,605,000 (12)	3,500,000 (16)	3,700,000 (12)	3,528,000 (12)	3,998,629 (12) 262K 60-Bit Words	4,326,148 (12)	4,936,868 (12)	6,183,270 (24)	4,031,325 (24)
Price/Mo With Maintenance (Lease Term In Years)	77,480 (4)	96,250 (4)	Not Available	73,969 (5)	82,888 (5)	83,565 (5)	95,350 (5)	123,386 (5)	99,779 (5)
Maintenance/Mo[1]	8,320	10,200	10,200	10,900	8,832	6,895	8,120	12,640	9,271
Memory Size In Megabytes (Minimum-Maximum)	4-8	4-8	4-16	3-6	131,000-262,000 60-Bit Words	6-16	8-16	1-8	2-16
Memory Cycle Time (Nsec)	290-464[11]	300	280	See Footnote 2	400[4]	600-100[7]	600-100[7]	See Footnote 10	See Footnote 10
Machine Cycle Time (Nsec)	58	32.5	28.5	Not Available	25	50	50	Not Available	Not Available
Channels (Minimum-Maximum)	12-16	16	12-16	56	12-24	12-104	12-104	56-224	56-224
Price/Channel	320,000 (4 channels)		150,000 (4 channels)	Not Available	85,650 (12 Channels and 4 Peripheral Processors)	54,075 (4 Channels or 1 Block Multiplexer)	54,075 (4 Channels or 1 Block Multiplexer)	Not Available	Not Available
Memory Buffer	64K	32K	32K	24K[3]	See Footnote 5	96K	96K	4 Buffers Of 2K Each	5 Buffers Of 2K Each

CW Chart

1. IBM maintenance is minimum monthly charge, 10 hour/day, 5 day/week. Amdahl and Burroughs rates cover 24 hour/day, 7 day/week.
2. Access time is 50 nsec/byte.
3. Consists of 24K program buffer plus data buffer and storage and central word register.
4. Effectively 50 nsec.
5. Scheduling arrangement for peripheral processor units and CPU to access main memory. Buffer could be one to 12 of CDC's 60-bit words.
6. Univac prices include software.
7. Speed is 600 nsec for 16 bytes in "rear memory" and 100 nsec from "forward memory" to CPU.
8. To achieve advertised performance levels close to Amdahl 470V/7, the 1100/84 should have 8M bytes of memory, bringing memory buffer to 128K.
9. HIS expects to offer other systems in this range based on Level 66 with Gcos operating system; the 68 DPS units have Multics operating system. The 68/80 is what HIS would have bid prior to October 1977; the 68/DPS is the firm's current offering.
10. Memory access time is 750 nsec.
11. Depends on type of work being done.

Computerworld 1978

price. The option removes a great deal of the risk involved in ownership of an untested system. The user has a chance to see his applications successfully run before he agrees to purchase the system. Lease with purchase option plan should be used if funds are available and if the system life, including reuse, is longer than five years, and if it is more economical and practical than the other ownership methods available.

4. Lease to Ownership

This plan is new on the computer procurement scene and is known by several different names: Special Lease-Purchase Plan, Alternate Payment Plan, Installment Purchasing, etc. These plans are all essentially the same; in that, monthly lease payments are made until some given number (generally sixty payments) have been made, or until some given amount (the purchase price of the system) has been paid, and then title of the computer passes to the lessee. Until that time the lessee has no obligation beyond a normal lease plan [Joslin 1977, Bucci 1973].

Lease to ownership plans are not offered by all the vendors since they are new. Rather, they are made available only upon request. Late in 1969, the Automatic Data Processing Equipment Selection Office (ADPESO) of the Department of the Navy started requesting that some form of a Lease to Ownership Plan be offered as one of the procurement alternatives in proposals submitted in response to the request for proposals issued by that office. After much ignoring of that request,

one vendor finally offered a Lease to Ownership Plan in place of offering a discount on the cost of his system. After that, several vendors offered similar plans and have continued to offer such plans on subsequent bids [Joslin 1977].

In order to make Lease to Ownership Plans an acceptable procurement alternative, three major problems have to be overcome:

1. The evaluation technique used in selecting the computer system must be broadened to consider value outside the stated system life,

2. The purchase alternative of procurement must be re-examined and reconsidered on more than just a cost basis,

3. The tax situation relating to 'gradual ownership' of a capital investment must be investigated.

A company's principle interest in Lease to Ownership plans might be due to their recognition of the advantages of ownership, coupled to an understanding of the unavailability of purchase funds within their present economic situation. If purchase funds are not available, then some form of Lease to Ownership Plan is essential if the activity ever wishes to own the system.

B. LEASING VERSUS PURCHASING

As in the case of most business problems, the purchase-or-lease problem should be settled on the basis of a careful consideration of many factors; both quantitative and qualitative. These considerations must be balanced against one another to come out with the optimal solution. One method is to consider the relationship between the cost of purchasing and the cost of leasing.

The quantitative factors are:

1. Cost involved in purchase versus lease arrangements.
2. The estimated useful life of the machine.
3. The desired rate of return on investment (ROI).

Where applicable, income tax benefits and salvage values should be considered. By comparing the cost of purchasing with the cost of leasing, the financial advantages of both methods may be studied. Quantitative analysis can assist management in deciding which method of acquisition to use. Methods of quantitative analysis are described in Table 13. Other costs and factors, such as insurance, risks of ownership, resale prices, income taxes, and so on, are disregarded since their effects can be easily included in the analysis when required.

1. Methods For Lease-or-Purchase Decision

a. Method I

This method compares the cost of purchasing with the cost of leasing without considering rate of return. Cumulative costs incurred, namely purchase cost in the first

TABLE 13

METHODS OF QUANTATIVE ANALYSIS

Purchase basis		Lease basis
Purchase cost:	$600,000	Rental charges for first six years (including maintenance charges) for the same usage as would be the case if purchased outright.........$200,000 per year
Estimated useful life:	6 years	
Maintenance charges:		
First three years:	$45,000 per year	
Last three years:	$55,000 per year	
Interest on borrowed funds (desired rate of return)	3%	

year and maintenance costs in the first year, second year, and so on are used. For the example in Table 14, the break-even point would occur just before the fifth year of usage. Thereafter, the purchase basis affords favorable financial advantages.

b. Method II

This approach considers purchase price, maintenance charges, and rental charges amortized over the useful life of the equipment (including interest costs) in equal amounts. This approach also, as with Method I, looks forward (accumulating technique) in analyzing the effects of purchasing outright or leasing. However, there are two basic differences:

1. The purchase price, maintenance charges, and rental charges amortize in equal amounts in time.

2. The interest (or rate of return) applies to the unamortized amounts during the period.

Both of these factors could be assumed to occur monthly or more often, but here the amortization and interest computations are assumed to occur annually. Since the rate of return would be computed on the book value of the investment, the interest (rate of return) for each year on the purchase price would be as follows:

First year:

$600,000 x 3%	$18,000
500,000 x 3%	15,000
400,000 x 3%	12,000
300,000 x 3%	9,000
200,000 x 3%	6,000
100,000 x 3%	3,000
Total	$63,000

TABLE 14

COMPARISON OF COST OF PURCHASING
WITH THE COST OF LEASING

Item of cost	1973	1974	1975	1976	1977	1978
	Costs by Fiscal Year					
Purchase basis:						
Purchase costs	$600,000	--	--	--	--	--
Maintenance charges (cumulative)	45,000	$ 90,000	$135,000	$190,000	$ 245,000	$ 300,000
Total purchase basis (cumulative)	$645,000	$690,000	$735,000	$790,000	$ 845,000	$ 900,000
Lease basis:						
Rental charges (including maintenance charges)	$200,000	$400,000	$600,000	$800,000	$1,000,000	$1,200,000
Difference:						
Purchase basis exceeds lease basis	$445,000	$290,000	$135,000	--	--	--
Lease basis exceeds purchase basis	--	--	--	$ 10,000	$ 155,000	$ 300,000

A formula for the above computation can be expressed as $\frac{n + 1}{2}$ (where n is the number of years), and can be derived using the formula for the sum of an arithmetic progression and the series, expressed as follows:

$$S = n/n + \frac{n - 1}{n} + \frac{n - 2}{n} + \frac{n - 3}{n} + \cdots\cdots + \frac{n - (n - 1)}{n} = \frac{n + 1}{2}$$

By substituting six years in the formula for n, the interest can be computed as follows:

$$\frac{6 + 1}{2} \times 3\% \times \$600,000 = 7/2 \times \$18,000 = \$63,000.$$

Similarly, a formula can be derived for computing interest on the maintenance charges and the rental charges; however, in each of these cases, the annual payments are considered individual investments when payment is made. Therefore, instead of one simple multiplier of $\frac{n + 1}{2}$ there are series of such amounts. The result is the following multiplier: $\frac{n}{2} (\frac{n}{2} + \frac{1}{2} + 1)$.

The application of the above multiplier would be as follows:

$\frac{n}{2} (\frac{n}{2} + \frac{1}{2} + 1) \times 3\% \times \$$ annual maintenance or rental charge (or incremental charges if there should be a variation in annual amounts)

As an illustration, the maintenance charges of \$45,000 under the purchasing alternative would result in the following interest (or rate of return) on the payments:

$$\frac{27}{2} \times 0.03 \times \$45,000 - \$18,225$$

This would be for six years. Since there is an increment of \$10,000 increase beginning in the fourth year, additional

interest (rate of return) would be \$1,350 computed as follows:

$$\frac{9}{2} \times 0.03 \times \$10,000 = \$1,350$$

The comparison of the two alternatives, using the above approach, would result in a decision to purchase the equipment.

To lease the equipment would mean an increase in cash outlay of \$298,425 over that of outright ownership of the equipment, computed in Table 15. The above computations assume that all of the amounts are amortized over the useful life of the equipment in equal amounts, except for the incremental increase in maintenance charges.

c. Method III

This method uses the discounting technique (present value method or discounted case flow) frequently used in other financial situations. It is illustrated by the schedule in Table 16.

The present value method shows the effect of including interest cost in addition to the other costs as shown in Method I. In contrast, the break-even point (indicating that the purchase method is financially advantageous) does not occur until after the equipment has been used for more than four years; i.e., sometime in the fifth year of usage. If other costs and factors are considered significant and they can be expressed in monetary terms, Method III permits an easy approach to the determination of the total financial advantage.

Since column (f) in Table 15 is a computation of the present value at the time the purchase-or-lease problem

TABLE 15

Purchase basis:				
Purchase costs		$600,000		
Interest on purchase costs		63,000	$ 663,000	
Maintenance costs:				
Cumulative for entire six years	$270,000			
Interest on $45,000 for six years	18,225	288,225		
Cumulative (increment) for three years	30,000			
Interest on $10,000 for three years	1,350	31,350	319,575	
Total costs to purchase outright				$ 982,575
Lease basis:				
Rental charges ($200,000 each year) for six years			$1,200,000	
Interest on rental charges for six years, computed using same approach for maintenance			81,000	
Total leasing costs				$1,281,000
Excess of lease costs over purchase costs				$ 298,425

TABLE 16

PRESENT VALUE METHOD
Incremental Approach

Year	Investment at beginning of year (a)	Rental charges less maintenance (b)	Interest cost 3% of (a) (c)	Recovery of Investment (b)-(c) (d)	Investment at end of year: Amount (a)-(d) (e)	Investment at end of year: Present value factor times (e) (f)
Purchase					$600,000	
1973	$600,000	$155,000	$18,000	$137,000	$463,000	$449,515
1974	463,000	155,000	13,800	141,110	321,890	303,412
1975	321,890	155,000	9,657	145,343	176,547	161,565
1976	176,547	145,000	5,296	139,704	36,843	32,735
1977	36,843	145,000	1,105	143,895	(107,052)	(92,344)
1978	(107,052)	145,000	(3,212)	148,211	(255,263)	(213,779)
		(note a)				(note b)

a-The net rental charges were used for this computation: first year $200,000 less maintenance charges under the purchase basis of $45,000 a year; second year, $200,000 less $45,000; and so on ($55,000 used for the fourth through sixth years).

b-Interest cost of 3 per cent was assumed to accrue during the year using the present value formula $1/(1+i)^n$: .97087379; .94259591; .91514166; .88848705; .86260878; and .83748426; respectively. Amounts were rounded. Negative values shown in parentheses denote financial advantage to the purchasing alternative.

arises (at the beginning of 1973), the financial advantage of purchasing the equipment under the present value can be calculated as shown in Table 17. The present value approach reduces the rental charges (and maintenance charges) to their present value for direct comparison with the purchase price, since the purchase price is already stated in terms of its present value.

Another approach using the present value method is the incremental approach which sets forth when the payout occurs, as shown in Table 18.

Table 19 shows present value factors, or conversion factors to convert future cash flows into present values. They are factors that when multiplied by an amount to be paid in the future, give the present discounted value of those funds. For example, at 6% interest, $1,000 to be paid in one year is equivalent today to $943.40; $1,000 to be paid in two years is equivalent today to $890, etc. If instead of money to be paid out, it is money to be received (or figured in tax deductions), then again a $48,000 tax savings a year from now (at 6%) is worth $48,000 x 0.9434, or $45,283.20 today. To give an idea, Table 20 describes the IBM 370/145 computer system configuration. The purchase column shows the manufacturer's new purchase price for each unit. The prices shown are typical and will vary depending on optional equipment and manufacturer's price changes.

TABLE 17

PRESENT VALUE METHOD
(Project Approach)

	Discounting Factor		Present Value
Purchase basis:			
Purchase costs: $600,000	1.00000000		$ 600,000
Maintenance costs:			
$45,000 a year times present value of an annuity of $1 for 3 years at 3%..............................	2.82861135		127,288
$55,000 a year times present value of an annuity of $1 for 6 years deferred for 3 years................	5.41719144	$297,946	
Less: $55,000 for 3 years.........	2.82861135	155,574	142,372
Total purchase price...			$ 869,660
Lease basis (including maintenance):			
$200,000 a year times present value of an annuity of $1 for 6 years at 3%..............................	5.41719144		$1,083,439
Lease basis exceeds purchase basis...............................			$ 213,779

TABLE 18

Year	Lease Costs	Mainte-nance Costs	Difference Favoring Purchasing	Present Value Factor	Present Value	Cumulative Present Value
1	$200,000	$45,000	$155,000	.97087379	$150,485.44	$150,485.44
2	200,000	45,000	155,000	.94259591	146,102.37	296,587.81
3	200,000	45,000	155,000	.91514166	141,846.98	438,434.79
4	200,000	55,000	145,000	.88848705	128,830.62	567,265.41
5	200,000	55,000	145,000	.86260878	125,078.27	692,343.68*
6	200,000	55,000	145,000	.83748426	121,435.22	813,778.90
Deduct: Present value of purchase price				1.00000000	600,000.00	600,000.00
Net present value (incremental approach) rounded (positive, indicating that purchasing would be financially advantageous)						$213,779.00

*Breakeven point (i.e., payout point in time)

TABLE 19

PRESENT VALUE FACTORS

Year:	0	1	2	3	4
Present Value Factor @ 6%	1.0000	.94340	.89000	.83962	.79209
Present Value Factor @ 8%	1.0000	.92593	.85734	.79383	.73503
Present Value Factor @ 10%	1.0000	.90909	.82645	.75131	.68301
Present Value Factor @ 12%	1.0000	.89286	.79719	.71178	.63552

Year:	5	6	7	8
Present Value Factor @ 6%	.74726	.70496	.66506	.62741
Present Value Factor @ 8%	.68058	.63017	.58349	.54027
Present Value Factor @ 10%	.62092	.56447	.51316	.46651
Present Value Factor @ 12%	.56743	.50663	.45235	.40388

TABLE 20

IBM 370/145 COMPUTER SYSTEM

Equipment Description	Purchase	Rental/ Monthly	Monthly Maintenance	True Rental (Less Maintenance)	Purchase ÷ True Rental
370/145 Central Processing Unit (256K)	$715,125.00	$14,932.00	$1,161.00	$13,771.00	51.9
1403-N1 Printer	34,350.00	885.00	199.00	686.00	50.0
2821-1 Control Unit	40,190.00	1,060.00	49.00	1,011.00	39.7
1419 Reader Sorter	124,470.00	2,714.00	252.00	2,462.00	60.5
2540-1 Card Reader/ Punch	32,930.00	710.00	124.00	586.00	56.2
2314-B1 Disc Storage Control	61,570.00	1,620.00	62.00	1,558.00	39.5
2319-B1 Disc Storage	38,250.00	1,000.00	210.00	790.00	48.4
3803-1 Tape Control Unit	34,430.00	900.00	106.00	794.00	43.4
3420-3 Tape Unit (3)	50,520.00	1,320.00	195.00	1,125.00	44.9
TOTAL	$1,131,335.00	$25,141.00	$2,358.00	$22,783.00	49.6

Tables 21, 22, 23, and 24 give the examples of Present Value Cash Flow Analysis for manufacturer's rental contract, third party lease (full payout), third party lease (non-full payout), and purchase (with zero residual value) respectively (assuming an IBM 370/145 configuration at $1,131,835.00) [Gustafson 1973, Szatrowski 1976, Fowler and Starr 1974, Randolph 1974].

TABLE 21

PRESENT VALUE CASH FLOW ANALYSIS FOR
MANUFACTURER'S RENTAL CONTRACT
(ASSUMING A 370/145 CONFIGURATION AT $1,131,835)

Year:	0	1	2	3	4
Rental Payments	---	301,692	301,692	301,692	301,692
Tax Savings	(37,728)*	(144,812)	(144,812)	(144,812)	(144,812)
After-Tax Cash Flow	(37,728)*	156,880	156,880	156,880	156,880
Present Value Cost @ 6%	965,680	152,439	143,810	135,670	127,990
Present Value Cost @ 8%	899,854	151,068	139,878	129,516	119,922
Present Value Cost @ 10%	841,050	149,748	136,134	123,758	112,506
Present Value Cost @ 12%	788,344	143,474	132,566	118,362	105,681

Year:	5	6	7	8
Rental Payments	301,692	301,692	301,692	301,692
Tax Savings	(144,812)	(144,812)	(144,812)	(144,812)
After-Tax Cash Flow	156,880	156,880	156,880	156,880
Present Value Cost @ 6%	120,745	113,911	107,453	101,380
Present Value Cost @ 8%	111,039	102,814	95,198	88,147
Present Value Cost @ 10%	102,279	92,981	84,528	76,844
Present Value Cost @ 12%	94,358	84,248	75,221	67,162

*Investment tax credit of 3 1/3%

TABLE 22

PRESENT VALUE CASH FLOW ANALYSIS FOR
THIRD PARTY LEASE: FULL PAYOUT
(ASSUMING A 370/145 CONFIGURATION AT $1,131,835)

Year:	0	1	2	3	4
Lease Payments	---	206,802	206,802	206,802	206,802
Insurance & Property Tax (1.2% of Value)	---	13,582	12,414	11,124	9,699
Maintenance	---	28,296	28,296	28,296	28,296
Pre-Tax Cash Flow	---	248,680	247,512	246,222	244,797
Tax Savings	(113,184)*	(119,366)	(118,806)	(118,187)	(117,503)
After-Tax Cash Flow	(113,184)*	129,314	128,706	128,035	127,294
Present Value Cost @ 6%	698,675	125,653	117,984	110,725	103,853
Present Value Cost @ 8%	645,889	124,523	114,758	105,703	97,306
Present Value Cost @ 10%	598,814	123,435	111,687	101,003	91,392
Present Value Cost @ 12%	556,406	122,385	108,759	96,600	85,751

Year:	5	6	7	8
Lease Payments	206,802	206,802	206,802	206,802
Insurance & Property Tax (1.2% of Value)	8,125	6,387	4,467	2,344
Maintenance	28,296	28,296	28,296	28,296
Pre-Tax Cash Flow	243,223	241,485	239,565	237,442
Tax Savings	(116,747)	(115,913)	(114,991)	(113,972)
After-Tax Cash Flow	126,476	125,572	124,573	123,469
Present Value Cost @ 6%	97,344	91,178	85,333	79,789
Present Value Cost @ 8%	89,619	82,296	75,594	69,374
Present Value Cost @ 10%	82,456	74,425	67,121	50,479
Present Value Cost @ 12%	76,071	67,435	59,731	52,858

*Investment tax credit of 10%

TABLE 23

PRESENT VALUE CASH FLOW ANALYSIS FOR
THIRD PARTY LEASE: NON-FULL PAYOUT
(ASSUMING A 370/145 CONFIGURATION AT $1,131,835)

Year:	0	1	2	3	4
Lease Payments	---	271,522	266,997	262,472	257,946
Tax Savings	(113,184)*	(130,331)	(128,159)	(125,987)	(123,814)
After-Tax Cash Flow	(113,184)*	141,191	138,838	136,485	134,132
Present Value Cost @ 6%	741,789	137,195	127,274	118,033	109,431
Present Value Cost @ 8%	687,062	135,961	123,792	112,679	102,534
Present Value Cost @ 10%	638,114	134,773	120,479	107,670	96,193
Present Value Cost @ 12%	594,183	133,627	117,321	102,975	90,357

Year:	5	6	7	8
Lease Payments	253,421	248,896	244,370	239,845
Tax Savings	(121,642)	(119,470)	(117,298)	(115,126)
After-Tax Cash Flow	131,779	129,426	127,072	124,719
Present Value Cost @ 6%	101,426	93,976	87,045	80,593
Present Value Cost @ 8%	93,273	84,821	77,110	70,076
Present Value Cost @ 10%	85,915	76,709	68,468	61,091
Present Value Cost @ 12%	79,261	69,504	60,929	53,393

*Investment tax credit of 10%

TABLE 24

PRESENT VALUE CASH FLOW ANALYSIS FOR
PURCHASE (WITH ZERO RESIDUAL VALUE)
(ASSUMING A 370/145 CONFIGURATION AT $1,131,835)

Year:	0	1	2	3	4
Purchase Payment	1,131,835	---	---	---	---
Depreciation	---	251,518	220,078	188,639	157,200
Insurance & Property Tax (1.2% of Value)	---	13,582	10,563	7,922	5,659
Maintenance	---	28,296	28,296	28,296	28,296
Pre-Tax Cash Flow	1,131,835	41,878	38,859	36,218	33,955
Tax Savings	(113,184)*	(140,830)	(124,290)	(107,931)	(93,754)
After-Tax Cash Flow	1,018,651	(98,952)	(85,431)	(71,713)	(57,799)
Present Value Cost @ 6%	679,902	(93,351)	(76,033)	(60,212)	(45,782)
Present Value Cost @ 8%	697,343	(91,623)	(73,243)	(56,928)	(42,484)
Present Value Cost @ 10%	713,302	(89,956)	(70,604)	(53,879)	(39,477)
Present Value Cost @ 12%	727,940	(88,350)	(68,105)	(51,044)	(36,732)

Year:	5	6	7	8
Purchase Payment	---	---	---	---
Depreciation	125,760	94,320	62,880	31,440
Insurance & Property Tax (1.2% of Value)	3,772	2,263	1,131	377
Maintenance	28,296	28,296	28,296	28,296
Pre-Tax Cash Flow	32,068	30,569	29,427	28,673
Tax Savings	(75,757)	(59,942)	(44,307)	(28,854)
After-Tax Cash Flow	(43,689)	(29,383)	(14,880)	(181)
Present Value Cost @ 6%	(32,647)	(20,714)	(9,896)	(114)
Present Value Cost @ 8%	(29,734)	(18,516)	(8,682)	(98)
Present Value Cost @ 10%	(27,127)	(16,586)	(7,636)	(84)
Present Value Cost @ 12%	(24,790)	(14,886)	(6,731)	(73)

*Investment tax credit of 10%

VIII. SOLICITING PROPOSALS

During specification preparation, thought must be given to the problem of soliciting and evaluating proposals. Soliciting proposals is no real problem; the problem comes in keeping control of the solicitation. The necessary contact with the vendors is usually a difficult thing to control. One or two vendors usually have managed to become a party of the "family". In fact, thoughts about the need for a (new) computer system probably were initiated by a vendor. Investigation of the system requirements developed by the system study team are more likely than not to have uncovered several equipment requirements that were unique to the inside vendor's systems. It is precisely because of items like this that vendor contact must be controlled.

The system requirements of the company are rarely in accord with the system capabilities of any one vendor's computer. The purpose of the acquisition is to find the computer system which most closely fulfills the system requirements (not to make the system requirements echo some given vendor's equipment capabilities).

The objectives of the vendor are not the same as the company's objectives; since the company will have to live with and pay for the selection it makes, the company's objectives should be met. The only effective way of assuring that the vendor's objectives are not dominating the company's is to remove most of the influence by removing the vendor.

At some period during the system study, vendors known to have computer systems which might be able to handle the system requirements should be asked to discuss their systems. Then all vendors should be "locked out" so that they no longer can influence the final system specifications. Up to the lock-out, one vendor usually will have exerted the most influence. The purpose of the presentations by several vendors is to reduce this influence and to demonstrate that other vendors' systems have desirable features.

The lock-out requires that one individual within the company be established as the sole point of contact with all vendors. This person should not be involved directly with the preparation of the system specifications. The lock-out should continue for the full period of the selection. The remainder of this section will concern itself with the problem of controlled solicitation and evaluation of the proposals. The review of the system requirements, covered earlier, is the basis of the solicitation, but there is more to a solicitation than just supplying system requirements.

The vendors could be asked to bid after being supplied with nothing more than the system specifications (requirements), a few statements about necessary vendor support and the due dates for the submission of proposals. Such a bid request might be sufficient, but it does not ensure effective vendor contact. Effective solicitation of proposals includes both a good specifications package and good vendor contact.

A. REQUEST FOR PROPOSAL (RFP)

A good RFP package should contain, at the very least, statements concerning the following twelve elements [Joslin 1977, Thrussell 1976]:

1. System requirements
2. Evaluation criteria
3. System support
4. Benchmark data
5. Bidders' conference dates
6. Check-in dates
7. Provision for handling questions
8. Proposal due dates
9. Vendors' demonstrations and presentations
10. Contracting conditions
11. Award dates
12. General comments

SYSTEM REQUIREMENTS. This section should contain the system requirements, as defined previously. Any limiting conditions that were uncovered during the study should also be included.

EVALUATION CRITERIA. The criteria by which the proposals are to be evaluated should be explained to the vendors. This includes both the factors to be evaluated and their relative values and is beneficial to both the user and the vendor. The vendor gains in three important ways:

1. By knowing the rules of the game the vendor is better able to decide whether he wants to participate. The decision

to play is an important one to the vendor, for while he realizes that the returns are high, he also knows that the entry fee is high. Before he can ever hope to be awarded the contract, he must prepare and submit a proposal. To prepare a proposal, a vendor might incur a cost of from $500 for a response to a simple hardware specification, to $20,000 or $30,000 for a proposal in which detailed system study is required, to over $100,000 for protracted studies of a multiplicity of systems which might additionally involve larger benchmark tests.

2. Knowing the user's evaluation criteria, the vendor has a much better understanding of what type of system must be proposed, and proposes accordingly. If the user identifies cost as having greater importance than run time, the vendor may tailor the proposal to a smaller system with fewer time-saving devices. If the user indicates, by value assignment, that reliability is important, the vendor can propose a system complete with error detection and correction features. Moreover, he can develop alternative proposals and determine with some degree of accuracy which alternative most nearly fits the user's requirements.

3. The vendor has a chance to comment on criteria, preferably early in the game, and identify areas in which conditions may not be fair or meaningful. It is always possible that the user may be willing to modify these conditions.

When the user discloses so completely the evaluation technique he plans to use, he benefits in the following four ways:

1. Only the proper vendors are bidding. Those whose system could not possibly win will see the handwriting on the wall and drop out. Overall, fewer systems may be proposed, but those proposed should all more nearly fit the user's requirements.

2. The need to evaluate a multitude of proposals should be minimal because the vendor would have been encouraged to discard inappropriate alternative proposals. The user is thus in a position to receive system proposals as nearly suited to his wishes as vendors can make them.

3. The user can now receive free expert advice on the stated system requirements and on the evaluation method to be used. Most vendor comments on the evaluation method will minimize the importance of features and abilities their equipment does not have and emphasize the value of those their equipment has. But also there will be valuable suggestions on better evaluation methods or in other meaningful areas that were not intended to be evaluated originally. Suggestions of this kind may help the users to get a system better suited to their desires and needs.

SYSTEM SUPPORT. The kind and extent of vendor support necessary for attainment of all system objectives should be stated, and may extend beyond maintenance and training needs. Programming assistance, special subroutines, and other special requirements for which the user has a genuine need, may be herein defined as prerequisite conditions.

BENCHMARK DATA. The benchmark programs to be used as system description and validation should be supplied, along with sample data and answers by the prospective user.

BIDDERS' CONFERENCE DATES. The bidders' conference is a formal presentation to the vendor, by the user, of his system requirements. This conference should be held a week or two after the vendors have received the specifications package and have had a chance to review the system requirements and desires. At the conference, any questions on the techniques to be used in evaluating the proposal should be discussed and resolved. The date for the conference and a general explanation of its purpose should be included in the request for proposal (RFP).

CHECK-IN DATES. Check-in dates are dates, determined by the user, on which each vendor should indicate whether or not he is still engaged in preparing a proposal; if so, if the vendor still has any questions, he should ask them at this time.

PROVISIONS FOR HANDLING QUESTIONS. Since questions will arise throughout the proposal period, the user should state provisions for handling them, not only at the beginning, but at the various stages of the selection.

PROPOSAL DUE DATES. The proposal due dates should be explained, along with a clear description of what will happen to late proposals.

VENDORS' DEMONSTRATIONS AND PRESENTATIONS. Some time after the submission of the proposals, the vendors should be

afforded the opportunity to tell why they submitted as they did. Having acquired benchmarks, the vendors should be required to provide demonstrations.

CONTRACTING CONDITIONS. The vendor should be informed that any promises he makes will be written into the contract and that the contract will have to be signed by a corporate official.

AWARDING AND DEBRIEFING DATES. Dates should be set for the awarding of the contract and debriefing vendors.

GENERAL COMMENTS. This section should contain any instructions on the format of the proposal, such as arrangement of information with the proposal, number of copies to be submitted, and so on. The purpose of this section is to make selection easier for the user by keeping things as uniform as possible among the several proposals.

The specifications package is the first official statement of the details that the vendor will have concerning the user's problems. Thus the specifications must be stated clearly, questions asked by the user should be meaningful, and the evaluation method to be used well defined. The specifications package establishes the vendors' first thoughts on the system. These thoughts are also important to the user because if the vendor is given to believe or suspect that the equipment request is of low quality, or that the system (or evaluation method) is poor, he may abstain from bidding. The vendor who decides not to bid may be withholding the system best suited for the task.

B. VENDOR CONTACT

Even with the lock-out policy, it will be necessary for the company personnel to come into contact with the vendors in several ways. First, there should be some time set aside when the vendors and the company personnel can sit down and discuss the specifications package. This will occur normally during a bidder conference. There also is a continuing need to answer the vendors' questions as they arise. The vendors should be allowed to make some form of presentation after submitting their proposals, and they should be required to provide any demonstrations called for in the specifications package. Finally, there will be the pleasure of telling some vendor that he has won and the necessity of telling the other vendors why they lost.

1. Bidders' Conference and Questions

A bidders' conference is not always necessary especially if the specifications are simple and straightforward. However, if a bidders' conference is to be held, it is essential that the user be well prepared for it. The best preparation is a good set of specifications with meaningful benchmarks. The evaluation procedure should accurately reflect the user's requirements in system capability. At the bidders' conference, the user should have his best people available to explain and define the requirements and evaluation procedures. The user must consider all questions with an open mind. Where there is a possibility that some requirement or

procedure might be wrong, the question should be held open until a proper answer can be found. Answers "off the top of the head" are not sufficient.

Any questions raised during the conference should be answered as quickly as possible. All the questions asked should be studied to determine whether they affect only the asking vendor or all vendors. If the latter, then the point should be clarified for all vendors.

The bidders' conference will not dispel all the vendors' questions; even if it did, others would soon appear. The best means of enabling vendors to ask questions is a telephone call to the asking vendor, then written documentation mailed to all vendors.

2. Vendors' Demonstrations and Presentations

When the specifications package proposals require that the vendors demonstrate their computer systems, the dates for these demonstrations should be left up to the vendors as much as possible. In running a benchmark program or other demonstration for validation purposes, the vendor may need to obtain the proposed components from several other systems being readied for shipment. After deriving the timing information required from these demonstrations, the vendor then must decide whether to release the components for shipment as planned, and perhaps not be able to reassemble the proposed system in time for an official demonstration, or to hold the components, thus tying up several systems which otherwise could be shipped.

If the vendor is permitted to demonstrate whenever he is ready, he can be spared such a difficult decision. The user also benefits by an early demonstration, which affords him an earlier check on the vendor. If things do not go according to the vendor's plans and the demonstration does not demonstrate quite what the vendor said it would, the vendor has an opportunity to take these findings into consideration when preparing his proposal. One or two weeks after the vendors have submitted their proposals, they should be given an opportunity to make a formal presentation.

3. Contracting and Debriefing

Normally many statements made in the proposal cannot be verified. Any which had a bearing on the winning proposal's having won should be written into the contract covering that system. Statements of this sort are those dealing with either the mandatory conditions or the desirable features requested. It should be made clear that the contract will require the signature of a corporate official. Without such a signature, the contract is no stronger than the position of the signer.

Salesmen, under a strong emotion, have been known to state anything! Corporate officials, when they are signing their names, are pledging their corporation's funds. This can be reinforced by penalty clauses contained in the contract which cover late delivery, failure to deliver, and the like. These penalty clauses, as well as the general writing of the contract, should be handled by the company's legal staff.

Immediately after a formal announcement of vendor selection, a debriefing should be arranged to ensure that none of the losing vendors is left for long in the position of knowing they lost, but not knowing why.

The debriefing is something that should be handled in private. Each vendor should be told exactly why he did not receive the contract. If the selection was handled openly, the vendor should have little dispute since he was aware of how his proposal stacked up. Usually, the major point of dispute will be centered on adjustments that were made to his proposal. If the vendor was made aware of, and agreed to, the adjustments during the validation phase of the selection, there should be little he can say. Table 25 gives the list of the factors which must be included in any contract [Thrussell 1976, Joslin 1977, Sabol 1972, Chorafas 1967, Webster and Johnson 1976, Yearsley and Graham 1973].

TABLE 25

ADP CONTRACTURAL FACTORS

Equipment inventory (model number, etc.)
Terms (rental terms or lease, etc.)
Performance
Delivery
Installation
Support
Maintenance
Warranty
Attachment (of other manufacturer's equipment)
Reliability
Training
Technical specification
Acceptance

IX. COMPUTER PERFORMANCE MEASUREMENT AND EVALUATION

Data processing can be viewed as a production facility which is to satisfy the needs of its users. Typical users may be the payroll department staff producing paychecks, programmers debugging programs, and engineers solving technical problems. The needs of these users are to have their jobs processed correctly, on time, and economically. System performance evaluation is an attempt to determine how well a specific system is meeting or may be expected to meet specific processing requirements at specific interfaces. This difficult task is more easily carried out as three distinct evaluation activities:

1. The Cost Activity. The objective of this activity is to determine the one time and recurring costs from the first planning stage through the replacement of the system.

2. The Judgment Activity. The objective of this activity is to evaluate the nonquantifiable factors such as:

What improvements can the vendor be expected to make in his product line during the next five years?

How will these benefit the company?

What level of system component maintenance can be expected?

3. The Performance Evaluation Activity. The objectives of this activity are to develop meaningful, quantitative measures of how the system may be expected to complete a day's work in a day and estimate the unused capability available.

The main interest of this chapter is only the performance evaluation activity, and it will be discussed in more detail in the rest of this chapter. Before system evaluation can begin three inputs are required:

1. The description of a specific system to be evaluated.
2. The specific processing requirements.
3. The identification of each interface across which the system is to be evaluated.

The first input demands that all the components of the system be specified prior to the start of the evaluation because the evaluation process is being applied to the entire system [Stimler 1974, Rosen 1976]. Every hardware device and precisely how that device is to be connected in the system must be specified. The operating system, user programs, and job scheduling procedures also need to be specified. This includes identifying when jobs will be run and which jobs are to be multiprogrammed. The accuracy of an evaluation depends directly upon how well the system is defined. When different configurations are to be evaluated, each must be evaluated separately.

The specific processing requirements must be identified for the second input. Typical of the information needed here is the work load to be processed and turn around times to be met by month, week, day and hour. Periods of heaviest processing loads and shortest turn around time requirements are of special importance. One processing requirement to evaluate would be the maximum number of transactions per hour the

terminal operator can enter and have the computer-generated outputs arrive within the required response and turn around times.

The evaluation interface between each user of the system and the rest of the system must be identified for the third essential input. It is usually a human-system interface, such as between a terminal user and the terminal device in evaluating a real time system.

The effectiveness of a system can also be described in terms of the capability to process a given workload, and the capability to meet time requirements of individual users. The efficiency is measured by internal delays and utilization of individual system components versus demand. Effectiveness measures are the prime performance measures. Values of these measures can be assessed from observations made at the external side of the evaluation interface: they are what is seen by the system users. These measures are frequently called external performance measures [Svobodova 1976]. Efficiency is an internal factor; values of efficiency measures usually must be obtained from within the system. These measures aid in identifying problems that diminish system effectiveness.

Examples of both external and internal performance measures are given in Table 26. Performance measures are most frequently expressed as mean values. In many cases, mean values are clearly inadequate measures of system performance. For example, if the variance of the response time of

TABLE 26-a

EXAMPLES OF PERFORMANCE MEASURES

Performance Measure	Description
	SYSTEM EFFECTIVENESS
Throughput	Amount of useful work completed per unit of time with given workload
Relative Throughput	Elapsed time required to process given workload on system S1/elapsed time required to process the same workload on system S2
Capability (Capacity)	Maximum amount of useful work that can be performed per unit of time with given workload
Turnaround Time	Elapsed time between submitting a job to a system and receiving the output
Response Time	Turnaround time of requests and transactions in an interactive or a real time system
Availability	Percentage of time a system is available to users

TABLE 26-b

EXAMPLES OF PERFORMANCE MEASURES

Performance Measure	Description
	SYSTEM EFFICIENCY
External Delay Factor	Job turnaround time/job processing time
Elapsed Time Multiprogramming Factor (ETMF)	Turnaround time of a job under multiprogramming/turnaround time of this job when it is the only job in the system
Gain Factor	Total system time needed to execute a set of jobs under multiprogramming/ total system time needed to execute the same set sequentially
CPU Productivity	Percentage of time a CPU is doing useful work (used as a measure of throughput)
Component Overlap	Percentage of time two or more system components operate simultaneously
System Utility	Weighted sum of utilization of system resources
Overhead	Percentage of CPU time required by the operating system
Internal Delay Factor	Processing time of a job under multiprogramming/processing time of this job when it is the only job in the system
Reaction Time	Time between entering the last character on a terminal or receiving the input in the system and receiving first CPU quantum
Wait Time For I/O	Elapsed time required to process an I/O task
Wait Time For CPU	Elapsed time required to process a CPU task
Page Fault Frequency	Number of page faults per unit of time

an interactive system is large, the user is likely to be dissatisfied with the system performance even if the mean response time is reasonably short. Thus, if the exact distribution of the response time is not known, at least the variance of the response time ought to be considered as a performance measure in addition to the mean response time. A good measure of performance of an interactive system is the percentile response time. N percentile response time is defined as the time limit that guarantees that the response times of N percentage of all requests are shorter than this limit [Sekino 1972]. One cannot expect the response time for very involved requests to be as short as the response time for trivial requests. Thus, response time (percentile response time) should be assessed separately for different classes of requests.

Performance measures can be specified only with respect to the type and the purpose of the evaluated system, its workload, and the purpose of evaluation [Svobodova 1976]. Performance measures must be well defined since they set a framework for the entire evaluation process.

Having selected performance measures, the crucial problem is to determine how these performance measures depend on the system workload and the system structure. An understanding of such a relationship is essential if performance optimization efforts are to be constructive, but it is also important when selecting a new computer system. An expression of this

relationship is the performance model of the system. The performance model is the ultimate goal of system analysis [Svobodova 1976; Rosen 1976].

The values of performance measures are determined by a combination of the following:

1. Measurement
2. Analysis
3. Simulation

The most accurate values are obtained when the system is measured under its real workload. Because of the variability or unavailability of the real workload, it is often necessary to design an artificial reproducible workload and measure the system performance against this artificial workload. Whenever evaluating a system that has not yet been implemented or is otherwise unavailable for measurement, it is necessary to develop a functional model of that system. The values of performance measures are then obtained either by analytical means or by simulation.

Measurement and modeling are complementary processes in that:

1. a model provides a framework for measurement,
2. measurement provides data for validating the model,
3. the model aids in testing hypothesis and finding solutions to performance problems, and
4. correctness of model predictions is finally verified by measurement.

A. THE SYSTEM PERFORMANCE EVALUATION METHODOLOGY

The system performance evaluation methodology requires the successful carrying out of the following six steps:

1. define the technical terms used,
2. establish performance criteria,
3. acquire the specific input data needed for each evaluation,
4. analyze the performance of the system being evaluated,
5. use appropriate evaluation aids, and
6. document the evaluation results.

Each step will be discussed in detail.

1. Define the Technical Terms Used

To be able to meaningfully answer the question "What is the performance of the system?" with "It is operating at 65 to 85% of its capability during the third shift", it is essential that both those asking and those answering the questions have a common understanding of all technical terms used, such as performance, capability, and system. Since there is no industry-wide accepted dictionary of data processing terms, different practitioners use the same word to have different technical meanings and different words to have the same meaning. If meaningful numerical expressions for performance are desired, this problem has to be overcome. The following guideline could be very satisfactory.

Start with the assumption that no English word or group of words has any inherent technical meaning. Performance, system, or time sharing assume technical meaning only

after those intending to communicate define each word or combination of words they intend to use. Further, the essential criteria for the definitions are:

a. They are clearly understood by those using them.

b. The definitions should be operational, i.e., permit physical measurement to arrive at numerical values.

c. New definitions should not be originated for commonly accepted terms which meet the first two criteria.

The first criterion of clarity is extremely difficult to achieve. The enormity of the task can begin to be appreciated when it is considered that the single word system is being used as a symbol to communicate the idea of a complex, operational organization of men, machines, programs, and procedures. Therefore, it is suggested that a definition which is adequate for effective communication should be developed and used at the time it is needed rather than wait for a perfect or a standard definition which might be available after the immediate need has passed [Stimler 1974].

The operational criteria require that such definitions as throughput, response time, and capability permit physical measurement to arrive at numerical values when the system is operational. The third criterion is intended to reduce the proliferation of different definitions for the same concept.

2. <u>Establish Performance Criteria</u>

A performance criterion is a performance standard with which comparisons can be made. For example, the criterion for the throughput of a system is here defined as the

total data processing work successfully completed during an evaluation period. Throughput, like every other criterion to be defined, is applicable to all classes of data processing systems. However, since the unit of data processing work is different for each class, the unit of work unique to the class of system being evaluated must be used.

3. <u>Acquire The Specific Input Data Needed For Each Evaluation</u>

Input data needed include the exact way each component is connected, the configuration and characteristics of all hardware and software components, and so on.

4. <u>Analyze The Performance Of The System Being Evaluated</u>

A "pencil and paper" analysis is the first essential level of analysis. This procedure is:

a. understand, in depth, the operation of the system,

b. understand, in depth, the operation of the system components,

c. set up a model of the system,

d. determine and keep in the model only the significant components,

e. derive mathematical relationships for each of the criteria to be used in the evaluation,

f. insert system characteristics into the mathematical relationships and derive the required results,

g. perform sensitivity tests which indicate the relative effect each component has on performance, and

h. prepare conclusions and recommendations.

5. Use Appropriate Evaluation Aids

Simulations, benchmarking, and resource utilization monitors are available aids for the evaluation process. These aids, properly used, can provide cost-effective supplements to the pencil and paper analysis. Improperly used, these aids can be expensive, time consuming, and misleading.

6. Document The Evaluation Results

One of the essential results of any performance evaluation and performance improvement effort is to document what was done, conclusions reached, and recommendations made. Documentation is an essential step in the methodology [Rosen 1976, Stimler 1974].

B. THE CONTROL OF SYSTEM PERFORMANCE

Listing all parameters that affect computer system performance would be an exceedingly difficult task. The performance of a computer system with respect to a specific application is a function of:

1. System configuration.
2. Resource management policies of the operating system.
3. Efficiency of system programs.
4. Effectiveness of the instruction set processor.
5. Speed of hardware components.

Performance characteristics are shaped in three stages:

1. system design,
2. system implementation, and
3. matching the system to a given workload.

Most of the performance evaluation and optimization efforts are presented in stage three, because each information processing system handles a different workload. This stage is concerned mainly with system configuration and resource management; that is, allocation and scheduling of Processor Memory Switch (PMS) components.

Performance of a particular computer system installation can be controlled in several different ways:

1. Adjustment of system control parameters,
2. Change or modification of resource management policies,
3. Balancing the distribution of load among system components through system reconfiguration (changes in the assignment of peripheral devices to channels or the assignment of files to physical storage devices, changes in the distribution of software components in the system memory hierarchy, etc.), and
4. Replacement or modification of system components.

As long as the user interface does not change, the system does not change to the user; only the performance does. However, configuration changes and software changes result in a new system, a system that has to be designed, analyzed, implemented, tested, and documented. Control parameters can be changed as needed without having to test the system operationally. Table 27 lists some system parameters that can be used to control system performance. Control parameters can be set either before the system is started, or modified if necessary during system operation [Svobodova 1976].

TABLE 27

EXAMPLES OF CONTROL PARAMETERS

Control Parameter	Description
Quantum Size	Time quantum in which the CPU of a time-sharing system is allocated to jobs
Internal Priority	Priority based on the demands of a job and services already received
Degree of Multiprogramming	Number of jobs that are simultaneously in the main memory and thus eligible to use the CPU
Memory Partition Size	Amount of main memory allocated to a single job
Window Size	Time interval for determining the working set of a job
Maximum Allowed Paging Rate	Maximum allowed paging rate in a demand paging system
Page Survival Index	Number of CPU bursts received by a program before an unreferenced page is removed from main memory
Number of Simultaneous Users	Number of terminal users logged onto the system
Device-to-Channel Assignment	Assignment of I/O devices to available channels

In the latter case, changes may have to be induced by the operator, or control parameters can be changed automatically in response to changing user requirements.

Turnaround time or response time measures not only the system performance but also the quality of the program that constitutes the job. Performance improvement with respect to a specific application ought to be approached from both sides: reducing the amount of work required by the application, and improving the efficiency of the system [Ferrari 1975; Hatfield 1971].

Sometimes improvement of system performance with respect to a particular performance measure is possible only at the cost of reducing performance with respect to some other measures. The qualitative value of a specific level of a performance measure is the user's preference for this level. Performance trade-offs can be resolved only if the relative preferences for different levels of different performance measures are known. Determination of the preferred conbination is the basic problem of decision theory.

The system performance can also be assessed in terms of the cost of using the system. The cost of using the system is a function of the system cost and the cost of the programmer. The higher the throughput, the lower is the system cost per unit of work. The shorter the response time, the less the programmer's time is wasted waiting for response and the lower is the cost of programming. As the system approaches its capacity (maximum throughput), the response time suffers.

A proper balance between throughput and response time has to be established such that the cost of using the system is minimized.

An important factor that influences the productivity of a system user is the ease of using the system for a specific application. This factor has received more attention under the label "human engineering". The response time belongs to the category of human-oriented considerations; however, it is neither the only important consideration nor the most important consideration. Ease of use and performance are frequently conflicting design requirements. Since both of these factors can make a user's task either satisfying or frustrating, there is no simple rule as to how to resolve this conflict [Svobodova 1976].

In general, several different system models are used during various stages of a performance evaluation project. These models can be divided into three general classes:

1. Structural models
2. Functional models

   Functional models, used in performance analysis, can be divided into four groups:

   a. Flowchart models
   b. Finite-state models
   c. Parallel nets
   d. Queuing models
3. Performance models

A structural model describes individual system components and their connections. Such a model provides a useful interface between the real system and a more abstract model. Structural models are most frequently represented by block diagrams. The level of detail in a block diagram can easily be varied since individual blocks can in turn be further laid down as self-contained block diagrams. Block diagrams generally show the paths of data flow as well as control flow, but they do not specify the conditions governing this flow. Thus, block diagrams are suitable only as the first general level description of the system under study.

A functional model describes how a system operates. It defines the system such that the system can be analyzed mathematically or studied empirically.

Flowchart models are suitable for studying program efficiency and execution time requirements. A flowchart model is a directed graph model where the nodes represent computational tasks and the arcs show the possible flow of control between tasks. Flowchart models of system components and users' programs can be used as building elements of a system model, tied together by a mechanism that simulates system resource allocation and scheduling [Anderson 1976].

A finite-state model can be used for analysis of utilization of computer system resources. A finite-state model can be represented by a directed graph and, in this case, the nodes represent the state of the system; the arcs represent

the transitions between states. The system state is composed of the states of individual system components and it thus relects concurrency of system operations [Coop 1971].

Parallel nets are directed graphs made of two different types of nodes: transitions and places. Places with arcs directed into a transition are the conditions that must be satisfied concurrently if this transition is to occur. Such nets were found to be a useful aid in the design and implementation of a simulation model and in a planning of measurement experiments.

In a queuing models concept, a computer system is a set of resources and queues for these resources. When a job enters the system, it is placed in one of the queues where it waits until the requested resource becomes available. After a request has been processed, a job either leaves the system or enters some queue again. Queuing models emphasize the flow of jobs through the system, but they also enable one to observe the state of the system. These models are the most widely used models in computer performance analysis.

A performance model formulates the dependence of performance on the system workload and the system structure. It is derived by analysis of a functional model for a specific model of workload.

C. CLASSIFICATION OF DATA PROCESSING SYSTEMS

Commercial data processing systems can be generally classified into three classes:

1. Batch systems

2. Real time systems

3. Interactive time sharing systems

Each class fits the definition of a data processing system in that it is an organization of hardware, software, user programs, procedures, and people capable of transforming specified inputs into specified outputs. However, from both the performance evaluation and the system design viewpoints, each is sufficiently different to require a separate classification. An essential difference among the classes is the unit of data processing work. Table 28 shows the characteristics used to determine the classification of a system [Stimler 1974; Rosen 1976; Wooldridge 1973].

D. THE UNIT OF DATA PROCESSING WORK

The unit of work for each class of system is briefly described in this section.

1. Batch Systems

The processing of a job, as identified in the job logging routine, can be used as the unit of measure for batch processing work. Jobs vary widely in the amount of input, processing required, storage used, and output generated. The characteristics of jobs may also vary during different parts of each day. The specific jobs and input data frequently vary with day of the week, week of the month, and month of the year. For evaluation it is necessary to determine a representative workload profile. In many batch processing facilities a full month is needed to process an acceptable representative work load. In engineering facilities a week may be enough.

TABLE 28

SIMILARITIES AND DIFFERENCES AMONG
BASIC SYSTEM CLASSES

General Characteristic	Class Characteristic		
	Batch	Real Time	Interactive Time Sharing
Availability of input data prior to the start of the run	Completely available	Not available	Usually not available
Importance of human intelligence in the processing loop	Negligible	Minor	Significant
Extent to which application programs are debugged	Debugged as far as is economical	Debugged as far as is economical	May be completely undebugged
Knowledge of system resources needed to process a job prior to start of processing	Almost completely known	Almost completely known	May be unknown
Status of the data base prior to start of processing	Completely known	Completely known	May be unknown

TABLE 28 (CONT'D)

General Characteristic	Class Characteristic		
	Batch	Real Time	Interactive Time Sharing
Turn around time requirements	From minutes to hours	Seconds	From seconds to hours
Relationship between input and output among jobs processed	Highly variable	Nearly fixed	Highly variable
Users on line waiting for output	Not on line	Always	Frequently
Characteristics of the workload	Individual jobs usually require widely different resources and processing times to complete	Individual transactions require about the same resources and short processing times	Individual jobs usually require widely different resources and processing times to complete
Ability to schedule jobs to maximize performance	Usually can be scheduled	Operations has no control	Operations has no control
System availability requirements	Moderate 90% - 95%	Usually very high - 98%+	Frequently high - 96% - 98%

2. Real Time System

The processing of a transaction is the unit of work for this class of system. The processing of a transaction includes the receipt of the input, its processing, and transmission of all required outputs. Each transaction is completed in seconds and there usually are a limited number of different transactions a user can input. There may be from two to fifteen different transaction types. The combination of the limited number of different transactions and the short processing time per transaction permits meaningful evaluation of real time systems in periods as small as ten to fifteen minutes. Inquiry and message are commonly used to denote a real time input and output. Transaction is used to include these terms.

3. Interactive Time-Sharing Systems

An interactive time-sharing system provides each terminal user with essentially all the system capabilities he would have at the computer console except that he must share the computer resources with other users. This capability means that one terminal user can be compiling and debugging a new program, another running a program for the first time, another building a new data base, and another generating complex inquiries of a data base. Some of these inputs require responses in seconds, others in minutes. Normal batch production jobs frequently are run in the background when resources are available.

The unit of work is the sum of transactions processed plus batch jobs processed. To make meaningful evaluations and comparisons it is necessary to use the identical mix of jobs for each calculation [Stimler 1974, Svobodova 1976, Borovits 1973]. Basic definitions of performance criteria are presented in Appendix A in the form of simple equations to facilitate the calculation of numerical values.

Appendices B and C of this document provide sample checklists for hardware and software respectively. They can be applied to computer systems or their major parts (CPU, peripheral units) to obtain some criteria for performance evaluation and comparison. The data obtained can also be used in the applications of the computer selection and evaluation methodologies.

X. CONCLUSION AND RECOMMENDATIONS

Today's great emphasis on computer systems is an important reason for management's increased concern about major expenditures. Talk of acquiring a new computer system causes considerable interest at many levels of management. Management has a major role in each of the three principal phases of acquisition: systems analysis and design, selection, and installation.

Management must appoint good people and support them for the acquisition effort for the new system. The individuals appointed to the acquisition team must be able to communicate with management to ascertain management's needs and desires. Management must also guide and direct the team. They should be informed as to the approaches that might be taken in acquiring the computer system. The establishment of realistic milestones must be required by the management.

The responsibilities of the management in the three phases of acquisition, mentioned above, can be stated as the followings.

A. In the systems analysis and design phase of acquisition, management:

1. should discourage pioneering with the new system,
2. should require systems life forecasting,
3. should demand system design alternatives,
4. should require meaningful economic justification,

5. should insist on common language programming,

6. must assure the availability of procurement funds,

7. must review the system requirements,

8. should require that system specifications (not equipment specifications) be issued to the vendors.

B. In the selection phase of acquisition, management:

1. must require competitive specifications.

2. should encourage the issuance of a presolicitation letter to assure that the competitive specifications sought in the previous step have been achieved. An advanced copy of these specifications should be sent to prospective vendors for their review. The vehicle for sending the specifications to the vendors is a presolicitation letter.

3. should require the establishment of a formal Selection Plan.

4. should insist that, for any medium to large scale procurement, representative benchmark mixes be used for workload representation and validation.

5. should require a formalized evaluation process for system selection.

6. should require the use of System Life Costing in the evaluation process.

7. should review the complete Solicitation Document and Selection Plan before the Solicitation Document is released to vendors.

8. must live by the Selection Plan.

C. In the installation phase of acquisition, management:

1. should require that all activities leading to installation be scheduled.

2. must assure that a formal system acceptance test is conducted.

3. must insist that thorough, complete documentation be provided.

The cost of the proposed system cannot be ignored, therefore when choosing a selection methodology basic elements must be observed. These are

1. the assessment of the value of vendors' offerings to the buyer (EVALUATION), and the

2. validation of the vendors' claims (VALIDATION).

Implementing sophisticated evaluation and validation techniques used for selecting large or medium-sized computers can easily tie up three or more people for one year or more. The cost of their time, travel, computer use, and other expenses can easily exceed $50,000. That expenditure may be justifiable for a $5,000,000 computer system, or even for a $500,000 one, but it makes little sense for the buyer to spend $50,000 to decide how best to spend another $50,000 for a small computer system.

The buyer must face the fact that it is necessary to invest a certain minimal amount just to play the computer selection game. He must know how he plans to use the desired computer system. Determining the need for a minicomputer may take one person six months and cost $10,000. The same task with respect to the need for a large computer may take a team of five people one year and cost $100,000.

In short, advancing technology has drastically reduced the cost of computer hardware, and at the same time inflation, coupled with the demand for more complex computer applications, has increased the cost of the human effort associated with computer services, namely, the system study, the selection process, programming, and operation.

If the buyer goes the competitive route because of regulations or otherwise, he should become concerned with ways to minimize the cost of the selection process. Since the cost of preparing (writing) the Request for Proposals or Solicitation Document is usually somewhat fixed and minimal (once a good document has been found for use as a model), only the evaluation and validation processes provide opportunities for reducing costs significantly.

Simplification should not be interpreted to mean that it is necessary to use an inferior selection methodology which:

1. May fail to consider all the desired (but not mandatory) items or features,
2. May not facilitate establishing meaningful and understandable relative values between the desirable items,
3. Does not permit disclosing the relative value of the desired items in the request for proposal (system),
4. Fails to incorporate systems life costing.

These failings can be avoided by using the simplified version of the Cost-Value or Requirements Costing evaluation methodology. The more time spent on establishing the values of the desirable features, the better the technique becomes.

In view of the fact that benchmarking is the only validation process which is really defendable, the cost of benchmarking must be cut. The high cost of benchmarking has always been at the top of any vendor's list of complaints about competitive bidding. The biggest cost factors to the vendors are the cost of debugging the benchmark programs and the cost of pulling together and holding a system of the type necessary to demonstrate the running of the benchmark programs. Benchmarking small systems rarely involves complex systems, so the only problem is that the vendors will not bid on small systems if too much is expected of them in debugging or running the benchmark programs.

To further simplify the validation process, demonstrating the benchmark mix can be required of only the winning vendor. This not only reduces the vendors' costs, but also greatly reduces the buyer's costs, for demonstrations quickly consume man-days and travel dollars.

The particulars (e.g., government/private sector, expected system cost, applications, etc.) of the buyer's situation must dictate the means to be used by him for simplifying the procedure for selecting a small computer which satisfies his needs.

APPENDIX A

PERFORMANCE CRITERIA

EQ. 1

$$\text{Throughput} = \text{Total data processing work successfully completed during an evaluation period}$$

EQ. 2

$$\text{Throughput rate} = \frac{\text{Throughput}}{\text{Wall clock system time (in hrs., min. or sec.) to process the throughput}}$$

EQ. 3

$$\text{Average throughput rate} = \frac{\text{Throughput of a representative workload}}{\text{Wall clock system time (in hrs., min. or sec.) to process the representative workload}}$$

EQ. 3 for real time systems

$$\text{Average throughput rate} = \frac{\text{Transactions successfully completed in a representative workload}}{\text{Wall clock system time (usually in min. or sec.) to process that workload}}$$

EQ. 3 for batch systems

$$\text{Average throughput rate} = \frac{\text{Jobs successfully completed in a representative workload}}{\text{Wall clock system time (in hrs.) to process that workload}}$$

EQ. 3 for interactive time-sharing systems

$$\text{Average throughput rate} = \frac{\text{(Representative short job workload + representative long job workload) successfully completed during an evaluation period}}{\text{Wall clock system hrs. expended to process that workload}}$$

EQ. 3 for real time systems multiprogramming background batch jobs

$$\text{Average throughput rate} = \frac{\text{(Representative transaction workload + representative batch workload) successfully completed during an evaluation period}}{\text{Wall clock system hrs. expended to process that workload}}$$

EQ. 4

Capability = Maximum achievable average throughput rate regardless of the timeliness of outputs over a given time period

EQ. 5

Operational capability = Maximum achievable average throughput rate while meeting timeliness requirements

EQ. 6

$$\text{Relative throughput rate} = \frac{\text{Average throughput rate for evaluation 1}}{\text{Average throughput rate for evaluation 2}}$$

EQ. 7

$$\text{Relative capability} = \frac{\text{Capability for evaluation 1}}{\text{Capability for evaluation 2}}$$

EQ. 8

$$\text{Percent unused capability} = \frac{(\text{Capability} - \text{average throughput rate})\ 100}{\text{Capability}}$$

EQ. 9

$$\text{Percent throughput rate change} = (\text{Relative throughput rate} - 1)\ 100$$

EQ. 10

Turnaround time = Elapsed time in hrs., min. or sec. between arrival or first character of first input at input interface and arrival of last character of final output required by that input at output interface

EQ. 11

Response time = Elapsed time in sec. between arrival of last character of an input transaction at input interface and arrival of first character of final output at output interface

EQ. 12

$$\text{Elapsed time multiplication factor (ETMF)} = \frac{\text{Turnaround time of a specific job processed in a specific multiprogrammed environment}}{\text{Turnaround time of the same job processed as the only job in the same system}}$$

EQ. 13

Equivalent throughput rate of N independent systems

= Sum of throughput rates of N systems

= Throughput rate of system 1 + Throughput rate of system 2 + ... + Throughput rate of system N

EQ. 14

Equivalent capability of N independent systems

= Sum of capabilities of N systems

= Capability of system 1 + Capability of system 2 + ... + Capability of system N

APPENDIX B

HARDWARE CHECKLIST

A. Central Processing Unit (CPU)

1. Organization (word and/or byte oriented)

2. Processor storage characteristics:

Real, buffered or virtual processor storage; core or monolithic; amount reserved for firmware; net amount available for operating system and problem programs. Amount of low-speed storage included, if any.

3. Complement of registers

4. Memory cycle time

5. Average "access to processor storage" time

6. Number of words or bytes accessed per cycle

7. Instruction repertoire

8. Instruction mix timing (average execution time)

   Example: (5-byte unpacked fields)

   a. c = a + b

   b. c = a + b

   c. c = a + b

   d. Move a to b

   e. Compare a to b and branch

Instruction mix should be chosen based on expected use. For instance, if a significant amount of floating-point work is expected, then these instructions should also be timed.

If the arithmetic instructions are performed in the registers, the loading to and storing from registers should be included in the timings.

9. Special power unit required

10. I/O channels

a. Number of channels by type (selector, multiplexor, or block-multiplexor)

b. Maximum speed of each

c. Attachable units (or excluded units)

d. Switching capability of attachable units

e. Simultaneity of operation between CPU and the I/O units, as well as between the I/O units themselves

f. In-board channel (CPU acts as channel processor) or out-board channel (channel processor separate from CPU)

g. Channel diagram of proposed system

h. Attachable to another CPU

11. Integrated controllers

a. Attachable I/O units

b. Limitations on which integrated controllers may or may not be core resident

c. Degradation of CPU performance caused by the integrated controllers

12. Timers/clocks

a. Resolution or precision

b. Maximum time accumulation

c. Interrupt triggers

d. Difficulty in setting

e. Time of day or interval timers

13. Power failure protection

    a. Emergency off-automatic shutdown sequence

    b. Power fail safe

    c. Standby or secondary power source

14. Storage protect capabilities

    a. Number of separate areas protected

    b. Fixed areas or software controllable

    c. Minimum area protectable

15. Compatibility/emulation features

    a. Machines emulated

    b. Software requirements

    c. Limitations

16. Expandability

    a. Other features available

    b. Maximum storage and channels

B. Magnetic Tape Units

1. Number of units

2. Number of controllers

3. Densities supported, single or dual

4. 7-Track/9-Track

5. Operating characteristics: Mounting operation (auto-load or manual), tape cartridge required or usable, fixed or rotatable dial, stress and wear on tape (number of capstans, vacuum column, tension arms)

6. Continuous or incremental recording

7. Transfer rate

8. Start/stop time

9. Rewind time

10. Formula for computing effective speed

11. Error-checking and correcting capability

12. Automatic or manual switching (between CPUs, channels, controllers)

13. Expandability: maximum number of units per controller, controllers per CPU

C. Card Read/Punch

1. Rated speed (reflects maximum speed)

2. Time to process one card (converted to cards per minute, this reflects minimum speed)

3. Card codes supported

4. Number of stackers and capacity of each

5. Number of hoppers and capacity of each

6. Error-checking capability

7. Buffered, interlocking or cycle steal

8. Special features: 51 column, punch-feed-read, mark sense, and so on

9. Capability for sorting, collating, interpreting (card print)

10. Noise level

11. Reliability

12. Controller characteristics and limitations

D. Printer

1. Rated speed (for designated character set)

2. Time to print one line

3. Number of print positions

4. Width of form (maximum and minimum)
5. Quality of print (single and multiple form)
6. Character set
7. Skip speed
8. Carriage tape specifications
9. Lines per inch
10. Noise level
11. Stacker characteristics
12. Reliability
13. Buffered, interlocking or cycle steal
14. Controller characteristics and limitations

E. Disk or Drum

1. Capacity
2. Transfer rate
3. Access time (seek and rotational delay)
4. Removable packs or fixed head storage
5. Special features (such as rotational position sensing)
6. Channel restrictions (such as attachable only to channel number one, or only device on the channel)
7. Controller characteristics and limitations
8. Expandability
9. Reliability

F. Operator Console

1. CRT or printer
2. Keyboard
3. Speed
4. Width of display

5. Number of display lines visible to operator
6. Character set supported
7. Location relative to CPU and I/O units
8. Noise level
9. Reliability
10. Special paper or stock form
11. Stacker for paper
12. Ribbon required-expected life

G. Paper-Tape Reader/Punch
1. Speeds (transfer rate, start/stop time)
2. 7- or 9-channel tape
3. BCD, EBCDIC or ASCII code
4. Feed and take-up reels or fanfold
5. Rewinding required
6. Checking capability
7. X-on and X-off required
8. Compatibility with source or destination of tape
9. Splicing considerations
10. Reliability

H. Telecommunications
1. Controllers (data adapters)
    a. Number of lines supported
    b. Speed of transmission
    c. Leased line or dial-up
    d. Synchronous or asynchronous
    e. Types of terminals supported
    f. Interchange code supported

g. Features supported (such as paper tape, answer-back, auto-call, multiple-record transmission, polling)

h. Buffered

i. Duplex or half-duplex transmission

j. Error correction/recovery

2. Modems - See above and below

3. Communication facility

a. Leased or dial-up

b. Multiplexed line

c. Duplex or half-duplex transmission

4. Terminal

a. Type of display (CRT or hard copy)

b. Input modes (such as keyboard or tape cassette)

c. Speed

d. Width of display

e. Number of lines visible to operator

f. Interchange code used

g. Special paper or stock form

h. Impact or thermal printer

i. Multiple copies

j. Paper-stacking facility

k. Intensity adjustment

l. Visibility of cursor

m. Error correction/recovery

n. Hard-wired or acoustic coupler

o. On-line or off-line transmission

I. Other Equipment

Many other types of equipment may be available to attach to or be used in conjunction with the computer system. Each requires various considerations regarding performance, suitability for the purpose, compatibility with other units, reliability, operator interface and physical characteristics. Listed below are some of these types of equipment:

1. Microfilm/microfiche
2. Plotters/graphics
3. OCR scanner
4. Array processor
5. Audio response
6. MICR
7. Manual or automatic switching units

Many other considerations such as power requirements, air conditioning, humidity control, floor space, and so on, apply to all the hardware.

APPENDIX C

SOFTWARE CHECKLIST

A. Operating System

1. Resident device(s)
2. Amount of direct-access storage dedicated to operating system and work space required
3. Processor storage reserved for operating system
4. Support for anticipated I/O devices
5. Extent of multiprogramming capability and limitations
6. Proposed method of card I/O and print processing (SPOOL)
7. Preexecution I/O device setup
8. Ease of operation
9. Acceptability of operator messages
10. Access methods available
11. Virtual storage-optional or required
12. Support of automatic switching between channels
13. Compatibility or emulation support-capabilities and limitations
14. Complexity and capability of job-control cards/language
15. Job-accounting facilities
16. Operating system and hardware performance statistics
17. Telecommunication facilities (Remote Job Entry, direct data entry/retrieval, time-sharing, and so on)
18. Clarity of error codes/messages

19. Data-base management features

20. Facilities for user program library

B. Compilers/Assemblers

1. Languages supported

2. Adherence to national standard languages and features

3. Processor storage required for execution

4. Work space required on direct-access storage

5. Maximum program size allowable (number of source statements)

6. Devices not supported

7. I/O addresses absolute or generic

8. Subroutine libraries available

9. Suitability of languages to meet expected needs

10. Telecommunication features

11. Clarity of diagnostic codes/messages

C. Sort/Merge

1. Maximum/minimum file size

2. Maximum/minimum record size

3. Fixed/variable record lengths

4. Blocking

5. Number of fields in key-maximum key size

6. Devices used/required/supported

7. Formulas/tables to compute processor storage and I/O storage required

D. Utility Program

1. List of utility programs available

2. Completeness of list to meet needs

E. Performance

1. Estimate sort timings

2. Estimate compile/assemble rate

3. Estimate operating system overhead

4. Estimate processing time of problem programs

5. Estimate compatibility/emulation performance

6. Predict total throughput of work load including operator functions and multiprogramming performances

7. Benchmark representative sample to confirm performance

8. Use of simulation where advisable

F. System Preparation Requirements

1. SYSGEN plan

2. On-site or remote

3. Minimum system required to perform SYSGEN

4. Amount of time required

5. Degree of testing needed

6. Vendor assistance

7. Education required

G. Software Availability/Reliability

1. How long in use by other installation (or when available)

2. Other users' experience

3. Software maintenance

a. Normal period between updates

b. Difficulty to maintain

c. Availability of vendor assistance

4. Quality and completeness of documentation

5. Computer program patent considerations

H. Vendor-Supplied Application Programs

1. Extent of library

2. Programs required

3. Programs not required but of potential value

BIBLIOGRAPHY

Auerbach Publishers, Inc., "Selecting a Small Business Computer," Data Processing Management, 1975.

Auerbach Publishers, Inc., "Requirements Costing: A Superior Computer Evaluation Methodology," Data Processing Management, 1975.

Auerbach Publishers, Inc., "Long Range Planning for EDP," Data Processing Management, 1976.

Auerbach Publishers, Inc., "Configuring the Computer System," Data Processing Management, 1977.

Auerbach Publishers, Inc., "Checklist for Evaluating Minicomputer Vendor Policies," Data Processing Management, 1977.

Basinoff, P., "Volume Buys Mean Volume Savings," Purchasing, p. 37, 18 December 1973.

Borovits, I., "Computer Selection and Acquisition," Data Processing, v. 17, no. 2, p. 98-100, March-April 1975.

Borovits, I. and Golembo, E. I., "Measuring Computer Performance," Journal of Systems Management, p. 38, November 1973.

Bucci, R. A., "Beware the Standard Lease," Datamation, p.75, March 1973.

Burroughs Clearing House, Burroughs Reports, v. 60, no. 11, p. 22, August 1976.

Chorafas, D. N., Selecting the Computer System, Gee & Co. Limited, 1967.

Clifton, H. D., Systems Analysis for Business Data Processing, Auerbach Publishers, Inc., 1969.

Computerworld Gazette, 16 January 1978.

Coop, D. H., An Analytical Approach to the Measurement, Evaluation, and Prediction of Computer Performance, 1971.

Coutinho, J. S., Advanced System Development Management, Wiley Interscience, 1977.

Data Center Dialog, v. 1, "Avoiding Power Outages - The UPS Option," February 1978.

Datamation, p. 59, "Comparing Costs on IBM Rental Plans," August 1972.

Datamation, v. 22, no. 1, p. 125, "Getting Around the Supply Schedule," January 1976.

Datamation, p. 186, "Should Happiness Come With a Price?," March 1976.

Dearden, J. and Nolan, R. L., "How to Control the Computer Resource," _Harvard Business Review_, p. 68, November-December 1973.

Department of Defense Computer Institute, _DODCI Course Catalog_, October 1976.

Doherty, W. J., "Scheduling TS5/360 for Responsiveness," _AFIPS Proc. FJCC_, p. 97-111, 1970.

Ein-Dor, P., "A Dynamic Approach to Selecting Computers," _Datamation_, p. 103, June 1977.

Eldin, H. K. and Croft, F. M., _Information Systems_, 1974.

Emery, J. C., _Can We Develop Cost-Effective Information Systems?_

Ferrari, D., "Workload Characterization and Selection in Computer Performance Measurement," _Computer_, p. 18-24, July/August 1972.

Ferrari, D., "Tailoring Programs to Models of Program Behavior," _IBM Journal of Research and Development_, v. 19, no. 3, p. 244, May 1975.

Fleet Materiel Support Office (UADPS-SP), _Time Analysis and Billing System (TABS)_, 1976.

Fleming, D. M., "An Accountant's Dream Come True," _CA Magazine_, v. 110, no. 3, p. 28, March 1977.

Fowler, M. H. and Starr, D. E., "Point of Indifference in Lease or Buy Decisions," _Financial Executive_, p. 8, March 1974.

Foy, N., _Computer Management - A Common Sense Approach_, Auerbach Publishers, Inc., 1972.

Gibson, J. C., _The Gibson Mix_, IBM TR 00.2043, June 1970.

Gustafson, G. A., "Computers - Lease or Buy?," _Financial Executive_, v. 41, p. 64, July 1973.

Hatfield, D. J., "Program Restructuring for Virtual Memory," IBM System Journal, v. 10, no. 3, p. 168, 1971.

House, W. C. and Shankle, J. E., "Choosing the Right Computer," Journal of Educational Data Processing, v. 3, no. 3, 1976.

House, W. C. and Shankle, J. E., Utilization of Cost/Effectiveness Models to Evaluate User Trade-Off in Terminal Selection, 1978.

Hunter, J. J., "Measure for Measure," Computer Decisions, v. 8, no. 9, p. 54, 1976.

Joslin, E. O., Application Benchmarks - The Key to Meaningful Computer Evaluation, 1965.

Joslin, E. O., Computer Selection, The Technology Press, 1977.

Krauss, L. I., Administering and Controlling the Company Data Processing Function, Prentice Hall, Inc., 1969.

Los Alamos Scientific Laboratory of the University of California, Evaluation of Computer Software.

Martin, J., Design of Man-Computer Dialogues, Prentice Hall, Inc., 1973.

McKinsey and Company, Inc., Unlocking the Computers' Profit Potential, 1963.

Naval Electronic Systems Engineering Center, ADS Development Plan, 11 January 1977.

Naval Supply Systems Command, Planning at NAVSUP.

Naval Supply Systems Command, "Automation of Procurement Accounting Data Entry, APADE, II," ADS Development Plan, 1977.

Nielsen, G. O., "How Maintainability Pays Off," Mini-Micro System, v. 9, no. 11, p. 42, November 1976.

Nolan, R. L., "Controlling the Cost of Data Services," Harvard Business Review, p. 114, July-August 1977.

Olson, G., "How to Choose a Bespoke System," Computer Weekly, v. 21, no. 524, p. 37, 18 November 1976.

Prokop, J., Computers in the Navy, 1976.

Randolph, J. M., "Computer Leasing Today," Financial Executive, p. 50, May 1974.

Raphael, H., "Simplify Low-Cost UP Selection with Tabulated Data," Electronic Design, v. 25, no. 5, p. 60, March 1977.

Reps, D., "Input/Output Cost Matrix," Hospital Financial Management, v. 31, no. 1, p. 31, January 1977.

Roenfeldt, R. L. and Fleck, R. A., "How Much Does A Computer Really Cost?," Computer Decisions, v. 8, no. 2, p. 77, November 1976.

Rogaw, B. J., "Should EDP Cost Be Charged Back to the User," Advanced Technical Training, IBM Corporation.

Rosen, S., Lectures on Measurement and Evaluation of the Performance of Computing Systems, Society for Industrial and Applied Mathematics, 1976.

Rothery, B., The Myth of the Computer, Business Books, 1971.

Sabol, T. A., "Purchasing a Computer," Management Accounting, p. 43, October 1972.

Schneidewind, N. F., Analysis of Computer Performance in Multiprogrammed Processing, 1975.

Sekino, A., "Performance Evaluation of Multiprogrammed Time Shared Computer Systems," Technical Report, MIT, Cambridge, Massachusetts, September 1972.

Sinnott, M., "Tune-up For Better Performance," Data Processing, v. 10, no. 6, p. 12, 1976.

Sippl, C. J. and Sippl, C. P., Computer Dictionary and Handbook, Howard W. Sams and Co., 1972.

Skyrme, D. J., "Computer Graphics - Part 1 - Choosing a System," Computer Aided Design, v. 3, no. 3, p. 18-23, Spring 1971.

Shope, W. L., Kashmarack, K. L., Inghram, J. W. and Decker, W. F., "System Performance Study," Proc. Share XXXIV, v. 1, March 1970.

Sollenberger, H. M., Management Control of Information Systems Development, National Association of Accountants, 1971.

Stimler, S., Data Processing Systems - Their Performance Evaluation Measurement and Improvement, Motivational Learning Programs, Inc., 1974.

Svobodova, L., "Computer System Performance Measurement," Technical Report, no. 66, Digital System Laboratory, Stanford University, June 1974.

Svobodova, L., Computer Performance Measurement and Evaluation Methods: Analysis and Application, Elsevier Publishing Company, Inc., 1976.

Szatrowski, T., "Rent, Lease, or Buy?," Datamation, p. 59, February 1976.

Tatham, L., The Use of Computers for Profit, McGraw-Hill, 1969.

Thrussell, C., "Purchasing A Computer," Computer Bulletin, Series 2, no. 10, p. 28, December 1976.

University of Michigan, A Performance Comparison of All-Digital Computation and the Advanced Hybrid Computer Systems, 1975.

Webster, G. J. and Johnson, C. W., "The Evaluation and Selection of a Computer System for Interactive Design," Computer Aided Design, v. 8, no. 4, p. 247, October 1976.

Withington, F. G., "Five Generations of Computers," Harvard Business Review, p. 99-108, July-August 1974.

Withington, F. G., "Beyond 1984: A Technological Forecast," Datamation, p. 54, January 1975.

Wolin, L., "Procedure Evaluates Computers for Scientific Applications," Computer Design, v. 15, no. 11, p. 93, 1976.

Wooldridge, S., Software Selection, Auerbach Publishers, Inc., 1973.

Yearsley, R. B. and Graham, G. M. R., Handbook of Computer Management, Gower Press, 1973.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman, Code 52 Department of Computer Science Naval Postgraduate School Monterey, California 93940	1
4.	Lt. Col. Ronald J. Roland Code 52RL Naval Postgraduate School Monterey, California 93940	1
5.	LCDR Stephen Holl Code 52HL Naval Postgraduate School Monterey, California 93940	1
6.	Lt. Nevzat Bayraktar VALDE CESME, SETBASI SOK. No. 3/20 MUTLU Apt. BESIKTAS ISTANBUL/TURKEY	4
7.	DENIZ K. K. Ligi Egitim Daire Bsk. BAKANLIKLAR ANKARA/TURKEY	5
8.	Maj. Jerry G. Paccassi II, USMC 395-B Ricketts Road Monterey, California 93940	1